zimory
going beyond...

HIGH AVAILABILITY IN CLOUD COMPUTING

August 2012

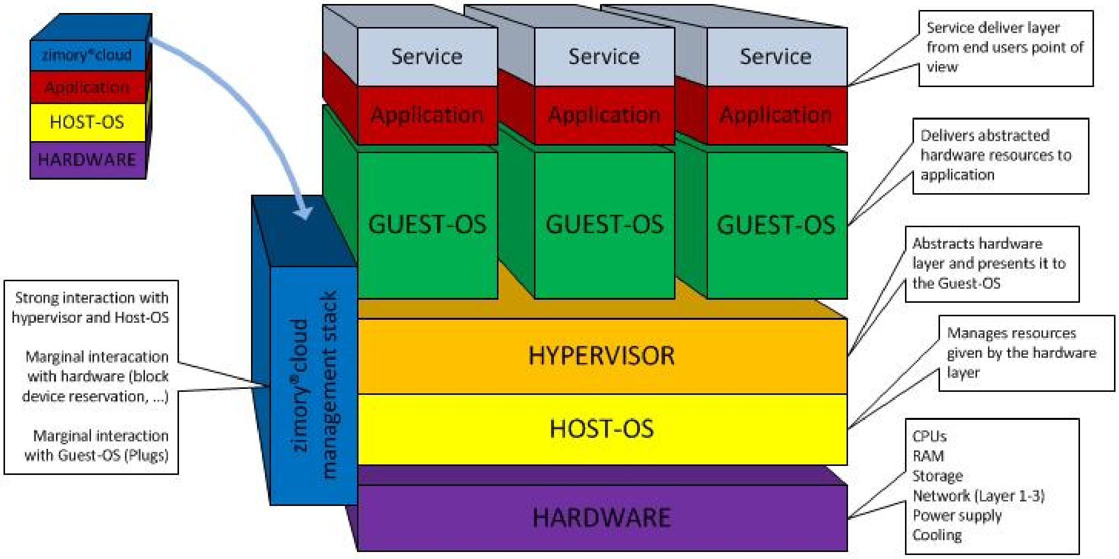
HIGH AVAILABILITY IN THE CLOUD

Defining the term High Availability is quite a complex task that can not be done in a few sentences. Depending on the addressed person, many possible definitions could arise for the HA definition question. There are many different and individualistic perspectives about its meaning. When asking an administrator for instance, he will probably provide an answer with regards to different clustering techniques, about redundancy or fault tolerant infrastructure. If talking to a back office manager, you will hear stories about broken printers and slow network shares. If the answer is provided by an architect, he will talk about missed deadlines because of lost documents and non-existent prints. All of them are speaking about the same thing, only from a different point of view. Thus and first of all, a global definition is needed in order to provide an overview of the aspects developed in this document.

For the purposes of this document, the term High Availability is defined as the capacity of an IT system to provide continuous service delivery. However, this definition rises another issue. What is the term Service referring to? For the purposes of Zimory, three aspects can be considered:

1. As the Zimory product is the cloud management stack, the service Zimory delivers is the possibility to start and stop, among many other use cases, a virtual machine inside the Cloud at any time.
2. As Zimory's customers are mostly providers of level 1, 2 or 3, their perspective of the service is to provide the hardware as well as the software so that virtualization can take place.
3. As Zimory's customers provide at their turm functionality to their own customers, the service is the application layer inside the virtualized machines.

Resulting from these considerations, there are three layers of availability to be considered: The management stack availability, the virtualization infrastructure availability and the virtualized application layer availability.



Service deliver layer from end users point of view

Delivers abstracted hardware resources to application

Abstracts hardware layer and presents it to the Guest-OS

Manages resources given by the hardware layer

CPUs
RAM
Storage
Network (Layer 1-3)
Power supply
Cooling

Strong interaction with hypervisor and Host-OS

Marginal interacation with hardware (block device reservation, ...)

Marginal interaction with Guest-OS (Plugs)

Figure 1. Layer Model for IT Systems

The graphic presented above shows the classical layer model for IT systems, including an additional virtualization layer representing the HOST-OS/HYPERVISOR element to be added to the Cloud Computing concept. From this perspective, zimory®cloud is not a layer in this model, but an additional, non-intrusive component. However, it is a service residing on the application layer, if< not in the cloud context itself.

The lower layer, where the hardware is to be found, describes everything the operating system or the hypervisor depends on. This includes CPUs, RAM or storage systems, regardless of whether it is a local hard drive or a remote SAN. For this layer to be available, redundancies are the most used technique, using for example, redundant power supplies or redundant SAN paths over redundant switches.

Going up to the next layer, the HOST-OS/HYPERVISOR layer, providing High Availability becomes a challenge since the functioning of every layer element is dependent on failure tolerance. It is up to the kernel not to crash because one of the underlaying pieces of hardware stopped working, and also to manage continuous operations and transparent failure handling for the above layers in such cases. Regarding virtualization, this layer also becomes, in a certain way, an application layer, the service provided is the communication stack between the guest-OS and the hardware layers.

The Cloud itself starts in the high up layers. The GUEST-OS layer is very similar to the HOST-OS/HYPERVISOR layer, but the services provided on GUEST-OS layer are intended to carry the application layer, which can be anything from an FTP server to a complex DBMS infrastructure, going beyond mere virtualized hardware.

With regards to continuous service delivery, the application layer above is the most important layer because of its flexibility, providing the highest number of possibilities for failure tolerant operations. There are as many HA concepts as there are enterprise class applications. It is also important to note that the cluster managers reside on this layer.

The cloud management stack is not seen as a discrete layer in the virtualization architecture, but as a plus to the hypervisor/host-OS layer. Regarding the software itself, it lives on top of the application layer and must be seen as a service. When designing High Availability cloud setups, it needs to cover two different ways of action. The first line is High Availability for all the layers doing virtualization, up to the application layer. On the other hand, the zimory®cloud dependencies have to be made highly available and fault tolerant in order to guarantee continuous delivery for management stack services.

From a technical perspective, every single layer must be secured with correspondent availability raising techniques. In addition, application layer support is a must in order to reach AEC-5, or even AEC-4 availability class. According to this fact, a cloud provider can only offer AEC-3 class for his virtualization infrastructure when the continuous service delivery is measured as uninterrupted operation of any single virtual guest. Otherwise, when defined as uninterrupted operation of at least 50% out of all virtual guests, reaching AEC-5 level is possible by applying industry standard HA techniques. With close collaboration with customers, who are responsible for the application layer inside virtual guests, it is also possible to meet those criteria for consumer services.

zimory®cloud does not provide load-balanced or hot-standby architecture deployment out of the box, but this functionality can be reached through the “Plug”-subsystem released in version 2.3.

HIGH AVAILABILITY METRICS

To measure, monitor and plan actions on building high available environments, many key performance indicators have been created. For the purposes of this document only two of them are relevant:

- MTTF (Mean time to failure): The value of this indicator shows, how long it takes before the service delivery is interrupted. Because there are no 100% available components to build IT systems, this value is always a positive, finite amount of time.
- MTTR (Mean time to recover): Shows how long it takes, to recover from a complete service outage back to normal operations. This value can be infinite, but is mostly a finite, positive amount of time.

THE THREE RS OF HIGH AVAILABILITY

Every electronic component is meant to fail at some point. There is no method to prevent failure at all, leaving only one alternative for providing High Availability for IT systems: the three Rs, the three Redundancies. Every physical part, on which the service delivery depends, needs to be doubled and even tripled in highly critical environments, since the failure of the first component causes losses in the system availability to handle further failures during the time it takes to replace or repair the failed component.

In most cases, the so called Hot spare clusters are used. However, it is also possible to use Cold spares, which can be used for non-vital systems or where the costs and effects of an temporary failure are not exceeding the costs of supporting a hot spare node. The time to recover from failures grows significantly with this method, but can also be adjusted according to individual needs and monetary possibilities. The most complicated method to provide redundancy is load balanced clustering. This is also the method the zimory®cloud product supports for providing high available cloud infrastructure, where every virtualization host is part of a load balanced cluster managed by zimory®central.

But not only the hardware needs to be redundant; it is even more important to keep the data highly available. The data availability layer is even more complex than the hardware, because data is rather more volatile than iron. Nowadays, it is also becoming common to operate with huge amounts of data, which complicates the problem even more. For continuous service delivery, it is absolutely necessary to have at least two synchronized sets of all data on-site to prevent loss from hardware failures like hard disk crashes. Furthermore, well-tested backup and recovery routines are requested in order to restore data deleted accidentally. In most cases, it is recommended to store those backups off-site, which leads to the need of having a third set of data for quick recovery on-site.

AVAILABILITY LEVELS

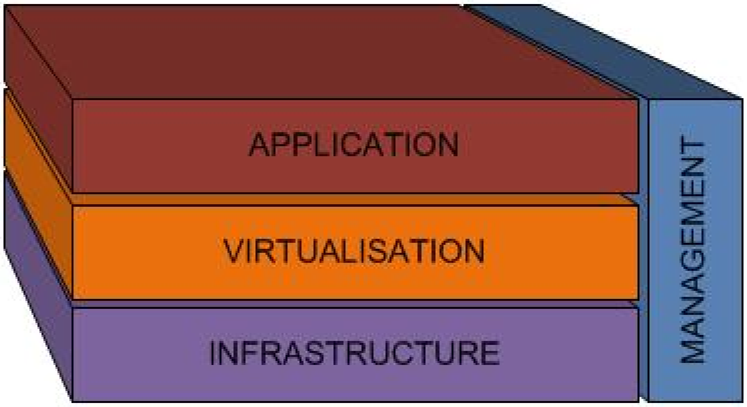
The following list presents the availability classification depending on the behavior of the system during system interruptions, maintenance and fault or disaster tolerance:

- AEC-0/Conventional: Function can be interrupted, data integrity is not essential.
- AEC-1/Highly Reliable: Function can be interrupted, data integrity must be however ensured.
- AEC-2/High Availability: Function may be interrupted only within fixed times or only for short time periods within operating times.
- AEC-3/Fault Resilient: Function must be maintained within fixed times.
- AEC-4/Fault Tolerant: Function must be maintained continuously.
- AEC-5/Disaster Tolerant: Function must be maintained under any circumstances.

DEFINING THE TARGETED LEVEL OF AVAILABILITY

Referring to the first chapter of this document, many levels require High Availability implementation. In addition, different perspectives must be considered when creating highly available cloud setups. Every single layer needs to provide a certain level of High Availability, but also the system itself needs to be handled as a whole and not just in parts, in order to provide a complete and coherent service delivery platform.

For zimory®cloud operators, there is a so-called Three Plus One Availability Model. This simplifies the system model presented above and allows the operator to separate the different High Availability techniques of their actual use.



Any cloud setup can be built as highly available as needed; measurements of any layer must work together. Before any work is done, the level of availability must be defined. Because availability is always a question of the proper price-performance ratio, the requirements must be well-analyzed. There is no need to build AEC-5 systems to support business processes, needed only from time to time without being performed at defined time schedules. On the other hand, it would be crucial for highly valuable processes with strict timelines and high workload to rely on AEC-1 systems.

Using this model allows to identify the required measurements to provide the needed availability on any layer or even on the whole system.

BUILDING HIGHLY AVAILABLE INFRASTRUCTURE LAYERS

The base of any cloud setup is the underlaying hardware, where every High Availability concept begins. Because of the wide possibilities and also the very different individual needs, it is not possible to create a common recipe for every environment. However, some basic rules can be defined as best practices and industry standards:

1. Defuse single point of failure. Every vital component failure that results in a complete or even partial interruption of service delivery needs to be identified and eliminated. Regardless of how reliable those

components are, they will fail at a certain point. In addition, components with unavoidable maintenance efforts and with a single point of failure lead to a significant decrease of the whole system availability class. Thus, every single component of this type must be redundant.

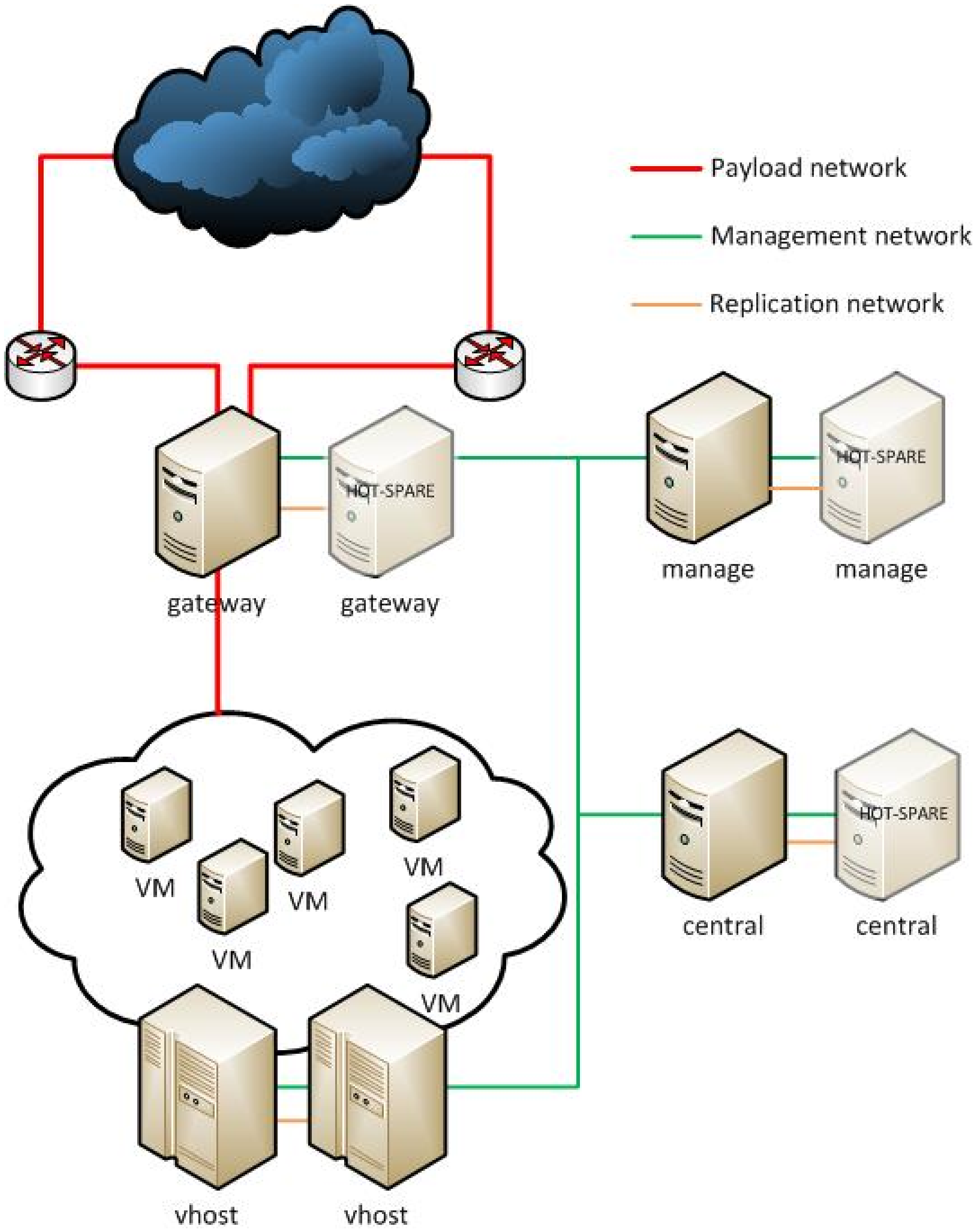
2. Use only well-proven components. Every vendor replaces his product lines with newer technology from time to time. Although every change is well tested and approved by the vendor, there is no proof that such changes can last on the long-term. For building highly available systems, it is wiser to choose veteran and mature products.
3. Avoid using components without clear support agreements. Every single component needs a well-defined support system from the vendor's for the whole lifecycle of the system. This includes long-term repairs as well as replacement support.
4. Choose fault tolerant components. Hot-plug technology for components is a must. Every single fan, hard disk or power supply needs to be replaceable without interruptions in the functionality of the component.
5. Keep and maintain replacement for vital components. If one of the cluster nodes fails, it must be replaced or repaired as soon as possible. While these replacements take place, the availability class of the whole system drops to the availability class of a single, stand alone component.

HIGH AVAILABLE ZIMORY®CLOUD MANAGEMENT STACK

Support of high available cloud management stack can be implemented by industry standard techniques, lowering costs and facilitating implementation.

The basic, High Availability setup illustrated in the following graphic, needs at least two hot-standby clusters: zimory®gateway and zimory®central, optionally a zimory®manage cluster can be provided, all of which are components of the cloud management system. The vhosts of the cloud operating system are generally inconvenient for keeping hot-standby nodes, since there are currently not stable, nor affordable technologies allowing to run virtual machines to spread over more than one physical machine.

In addition, every piece of infrastructure needs to be doubled. Redundant switches, redundant routers and redundant upstream connection as well as redundant power supply are required.



Although applying those techniques doubles the hardware acquisition costs, it also raises the availability of the management stack from AEC-1 to somewhere between AEC-3 and AEC-4, depending on the quality of the used HA hardware and software.

It is also possible to build a cluster setup of any zimory®cloud component standard techniques as provided by any of the compatible applicable Linux distributions. This is translated in most cases, into the use of a cluster resource manager such as pacemaker from the Linux-HA project. The deployment and configuration mechanisms are integrated into distributed OS management when using enterprise Linux like RHEL or SLES. Otherwise, it needs to be manually installed, a process does not concern this document.

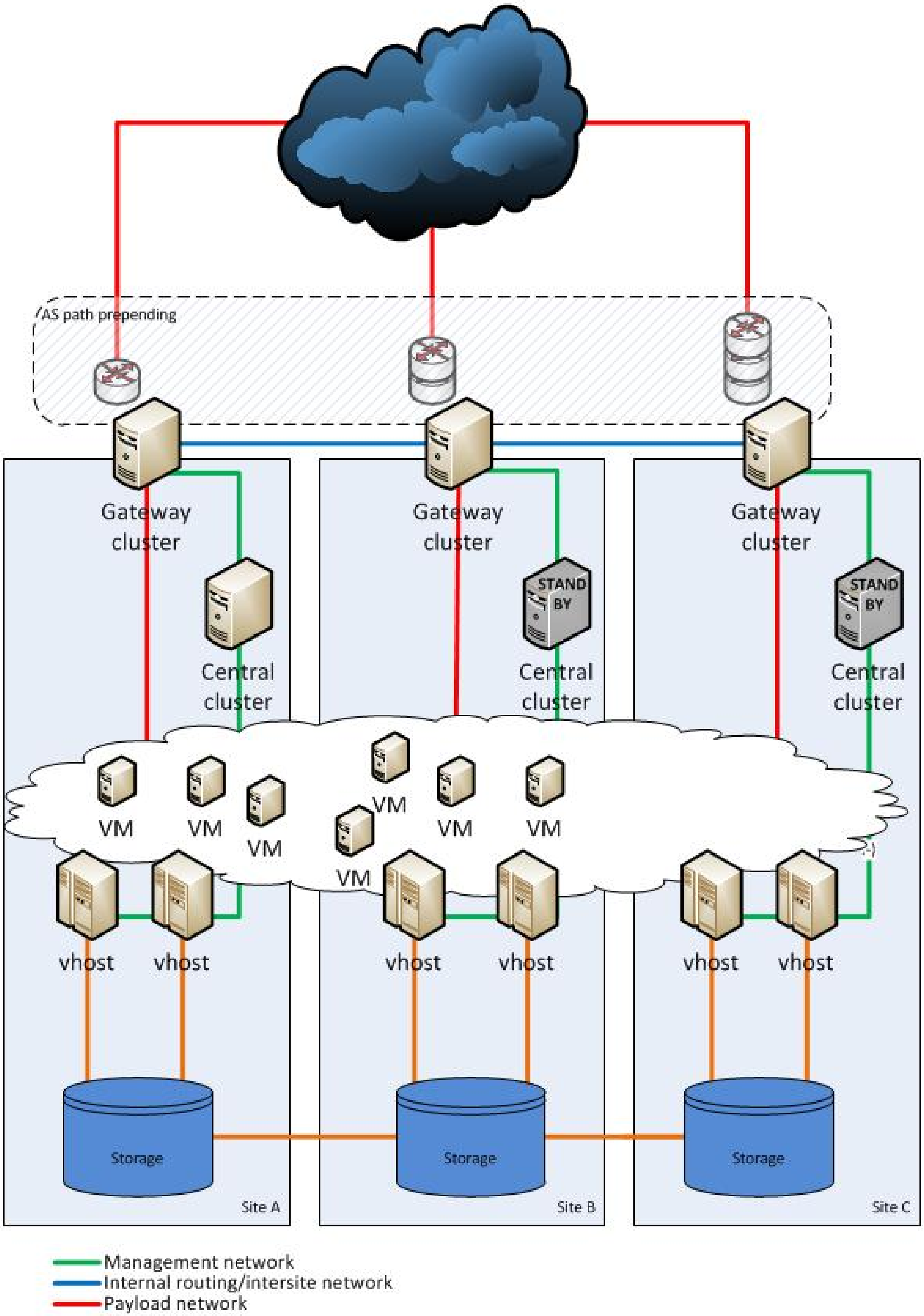
The important fact to keep in mind is that not only resource monitoring and management is needed, but also synchronous data replication between single cluster nodes. DRBD mirroring can be used for build-in disks (or RAID-Arrays), which is the cheapest, although the slowest solution. There is also the possibility of implementing one of the proprietary mirroring methods as provided by almost all storage vendors when using SAN or NAS storage technologies. It is even possible to use one single storage for both cluster nodes. The method choice will depend on individual needs and possibilities. Because the processes used to manage the zimory®cloud are not I/O extensive, they work regardless of the used technology.

Try to avoid database synchronization using the built-in of the given DBMS (like MySQL binary replication). This replication is done on the application layer and there are thus significant concerns about the reliability and performance of such methods. Furthermore, performing this replication generates a broad set of problems regarding disaster recovery and fail-over scenarios.

DISASTER-TOLERANT ZIMORY®CLOUD SETUP

Disaster scenarios for IT systems are mostly provoked by a long chain of non-critical failures resulting in a complete failure of service providing mechanisms. However, there are also much more trivial but dreadful causes, which can knock-off even the most resilient setup; these uncontrollable forces could be for example, flood, fire or wind. There are not enough prevention technologies for every possible case, the only way to solve these problems is by avoiding them, implementing effective prevention methods. This means, for example, to spread the system geographically, lowering the probability of disaster impact to the whole system and reduce exposure to eventual hazards only to some parts of the system.

Nevertheless, the costs of needed measurements to reach the first level of disaster tolerance nearly triples for a non-disaster tolerant setup. The good news is that considering the layer on which such tolerance is provided, the overall cloud performance rises. Thanks to zimory®cloud scaling methods, off-site resources can be used as a productive part of the whole system. However, further considerations of such setups must be considered in order to overcome resource shortages in disaster scenarios as it was previously explained.
The following illustration is an example of a disaster tolerant setup with three independent data centers.



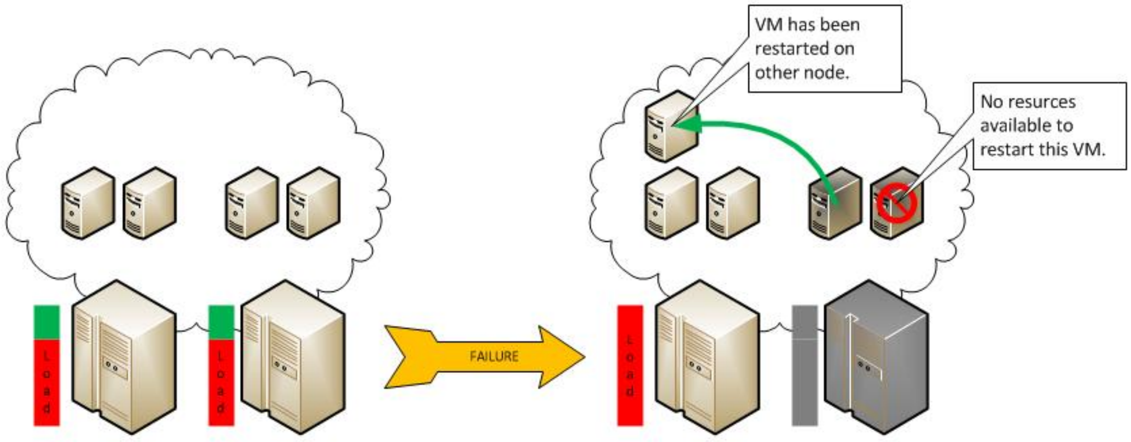
This kind of setup includes three independent data centers, each of them providing a complete set of all required components. In case of disaster, every single site is able to operate in a stand-alone mode. For normal operations, it is also possible to transparently share the virtualization cloud. The setup presented above includes mirrored storages with an high-speed, low latency interconnect, as well as an independent routing interconnection between the gateways. By applying dynamic routing technologies such as BGP with AS prepending, a data center black out can be almost completely transparent to the service consumer.

Unfortunately, for the time being, there is no direct support for such setups including those of zimory®cloud, leading to the inclusion of additional cluster and site management layers. This can be done with nearly any proprietary products suitable and with the implementation of freely available technologies from the Linux-HA project.

HARDENING THE VIRTUALIZATION LAYER

RESOURCE SHORTAGE ON PARTIAL FAILURES

In case one or more of the virtualization hosts goes down, every single virtual machine assigned to this hardware will disappear and needs to be restarted on functional hardware. From time to time, especially in well utilized environments, it can become impossible to provide the needed resources for those machines. In that case, it is not possible to recover from such failure as long as the failed hosts are not replaced.


VM has been restarted on other node.
No resurces available to restart this VM.
Load
Load
FAILURE
Load

To avoid this condition, the cloud provider needs to take several steps. First of all, a priority based approach to this problem is to be "established", which means that all machines need to be tagged depending on their importance. This model allows the system to be aware of the importance of the started machines, and provides the possibility to stop less prioritized guests in order to start high priority machines at their place.

In a second step, if needed, the remaining resources can be reallocated between the high priority guests that are still running. With most of the hypervisors currently available, it is possible to change the amount of memory or the CPU scheduling on runtime, sometimes even without noticing the guest OS. This gives the system the possibility to maintain operations without significant interruptions until it can be recovered to normal state.

Unfortunately, none of these methods are implemented into the zimory®cloud management stack and therefore, needs to be done differently. The most common way is to use an external monitoring host, that detects which virtual guests crashed because of hardware failure. This monitoring system calculates the needed resources to restart those machines and decides, if other machines need to be stopped. All the needed information is provided through the zimory®cloud REST-API, and every mentioned action can be taken through it as well. The resource reallocation scenario is for the time being in a highly experimental state, but it is possible (with some workarounds), to provide such recovery methods in worst case scenarios.

USING SLA DEFINITIONS FOR HA PURPOSES

zimory®cloud has a build-in system for prioritizing virtual machines according to the chosen SLA. By design, it is meant to provide the possibility of mixing in one cloud highly available hardware with standard hardware. This functionality can be used as a hook for HA procedures and tagging.

The first aspect of this functionality is to recognize high priority deployments in shortage scenarios caused by partial failure. At some point, there will be not enough resources to run all started machines, or to restart machines crashed because of hardware failure. If machines categorized with “Gold” level, previously running on a now crashed host, cannot get needed hardware to run, “Bronze” (and even “Silver”) guests can be stopped. This allows free resources to be assigned to high class guests.

Examples of SLA Definitions

The following table presents zimory®cloud SLA definitions according to the power supply, network, storage and availability types:

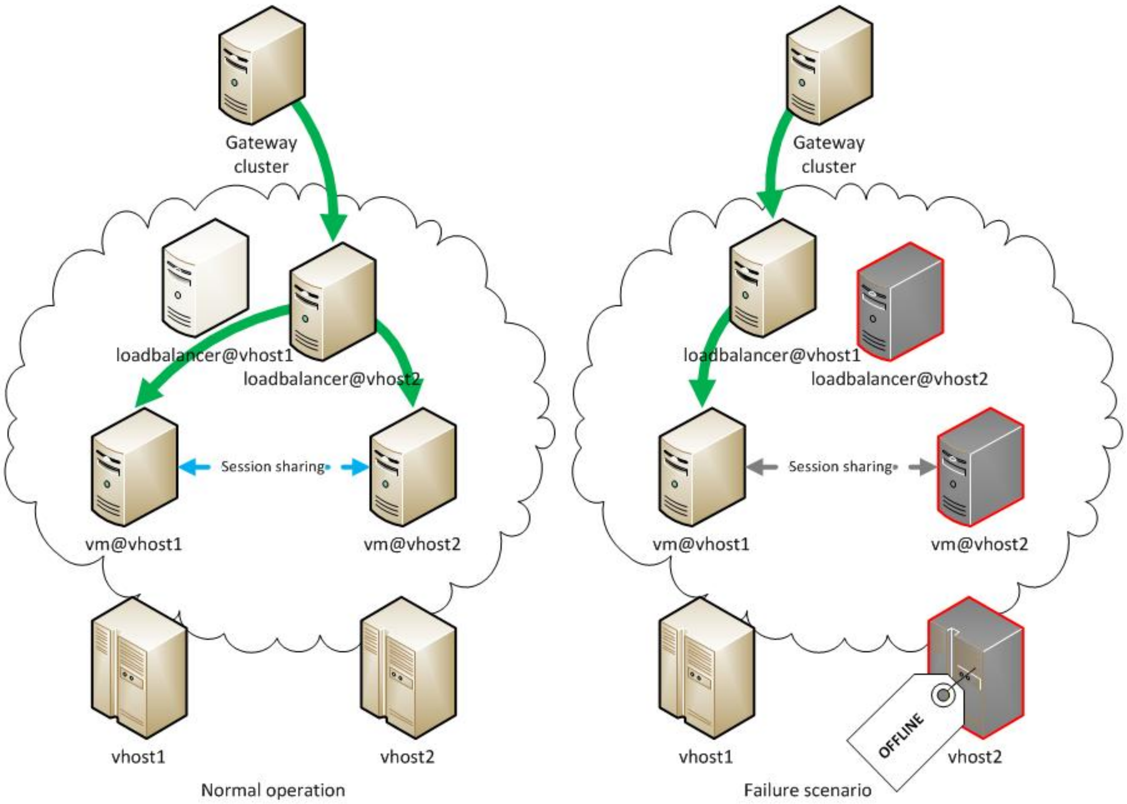
SLA Model Name	Infrastructure definition	Infrastructure availability	Management GUI availability
Gold	Power supply: Redundant, battery buffered Network: Redundant, QoS Storage: min. RAID5, off-site mirroring, daily snapshots Availability: Guests running on guaranteed resources.	99,999%	99%
Silver	Power supply: Redundant Network: Redundant Storage: min. RAID10, on-site mirroring Availability: Guests can be switched-off in case of resource shortage caused by multiple hardware failures.	99,99%	99%
Bronze	Power supply: Single Network: Single Storage: min. RAID1 Availability: Guests can be switched-off in case of resource shortage caused by single hardware failure.	99,9%	99%

Gold-Zone-A	See "Gold". A Gold-Zone-A guest will not be started in same site as a Gold-Zone-B guest.		
Gold-Zone-B	See "Gold". A Gold-Zone-B guest will not be started in same site as a Gold-Zone-A guest.		
Silver-Zone-A	See "Silver". A Silver-Zone-A guest will not be started on same hardware as a Silver-Zone-B guest. Both zones can, but must not, reside in the same data center.		
Silver-Zone-B	See "Silver". A Silver-Zone-B guest will not be started on same hardware as a Silver-Zone-A guest. Both zones can, but must not, reside in the same data center.		

PROVIDING A HIGH AVAILABLE VIRTUALIZED APPLICATION LAYER

The last step to highly available clouds must be met directly on the application layer within the virtual machines. It is essential to build virtual clusters, because any single interruption on the virtualization layer itself causes drop out of virtual machines. Although the virtualization service itself is not affected (any single virtual guest can be restarted at any time), the application layer operations depend on the virtualized operating system, which needs to be restarted. Such reboots generally take only severe time in a dimension measured in seconds, but this drops the service availability all the way down to AEC-2 level.

The approach to this problem is very similar as standard HA setups without virtualization. Hot standby or load balanced cluster technology can be used in most cases. Since the zimory®cloud system does not provide any particular solutions to this problem, additional provisioning for virtual guest setups are to be done.



The graphic presented above shows a standard load-balanced cluster setup inside the cloud. Using such techniques is the only possibility to raise the application layer availability above the AEC-1 level.

There are possibilities within any zimory®cloud setup to semi-automate the deployment procedures for this kind of virtual clusters. Thanks to the "Plug"-Subsystem, most of the customization can be applied from within the user interface. Obviously, any thinkable clustering technology can be supported this way.

KEEPING THE DATA HIGHLY AVAILABLE

Any single business process that creates value needs some kind of data to work with. Accordingly, the uninterrupted availability of this data is a major concern when developing highly available cloud systems.

There is a very large set of technologies and solutions for providing failsafe data storage. The choice depends on individual needs and of course, the price the customer is willing to pay. This chapter intends to give a short overview about industry standard technologies and their integration into zimory®cloud.

On the other hand, there is an urge to parallelize access to data. In active-active or load-balanced environments this is a highly sophisticated problem, that rises particularly when trying to provide transparent write-access for multiple virtual guests. Most solutions still depend on sharing-aware systems in the cloud, and there are no production solutions available to safely overcome this limitation.

Every shared storage needs to implement their own High Availability methods in order to lower the “single point of failure” risk. An overview of all those methods would go beyond the goals of this document. However, the basic techniques behind those methods are:

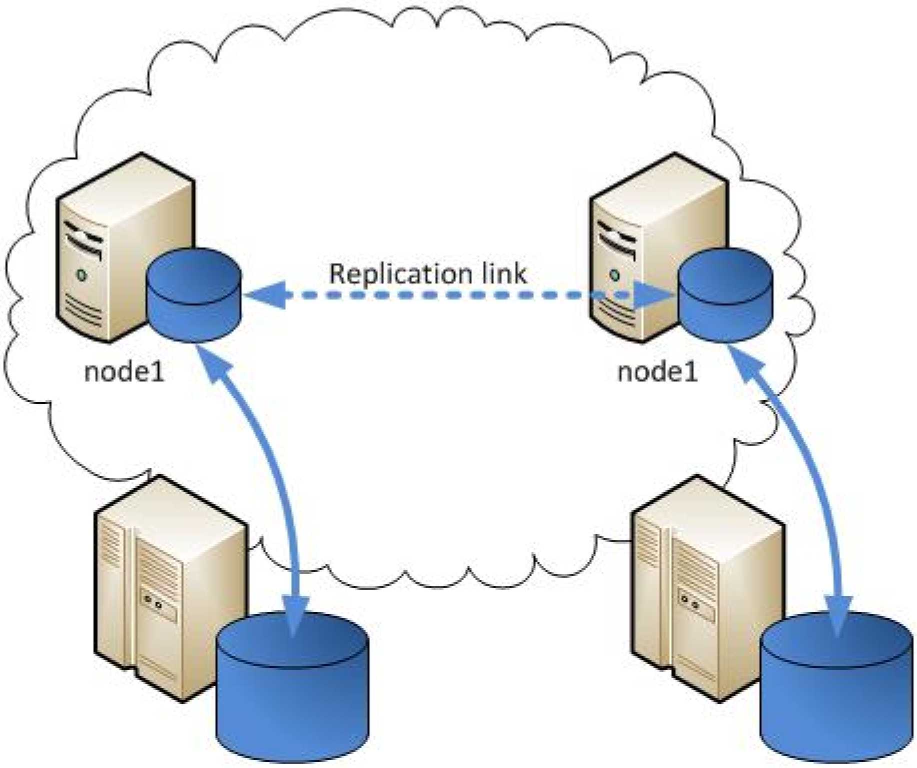
- RAID: This technology implements High Availability on the physical hard drive layer. Multiple disks are combined into logical devices. The higher the count of disks and the more the data is spread, the lower the possibility of data loss on multiple disk failure. RAID arrays can be done in hardware as well as in software, where most modern operating systems can emulate any possible RAID configuration. Also any enterprise class storage system combines many different RAID techniques to provide high reliability together with high performance.

- Mirroring: The data is mirrored between two or more storage nodes to prevent data loss in case one of complete failure of one of the nodes. In addition, it ensures continuous operation in case of hard disk (or RAID array) failure. Mirroring can be done either synchronously or asynchronously, block or file based and as two-node or multi-node. The most common scenario is active-passive two-node mirroring, where only one part of the mirror is writable. Active-active setups unconditionally require the usage of a cluster file system. Multi-node setups are mostly used for impact-free backups or follow-the-sun scenarios.
- Multipathing: Client-based technology for ensuring continuous operation in case one of the available, mirrored storage nodes or any piece of infrastructure between client and storage node fails. This technology can also be used to raise the throughput between storage and client.

For enterprise class storage systems, all of these technologies are combined into monolithic products.

There are many methods to integrate storage into the cloud. Some of them are described in the following sections of this document.

SHARED NOTHING, CLOUD-BASED SOLUTION



The cheapest and quickest way to deploy high available storage is done by using shared-nothing mirroring solutions managed by applications and/or operating systems running inside virtual machines.

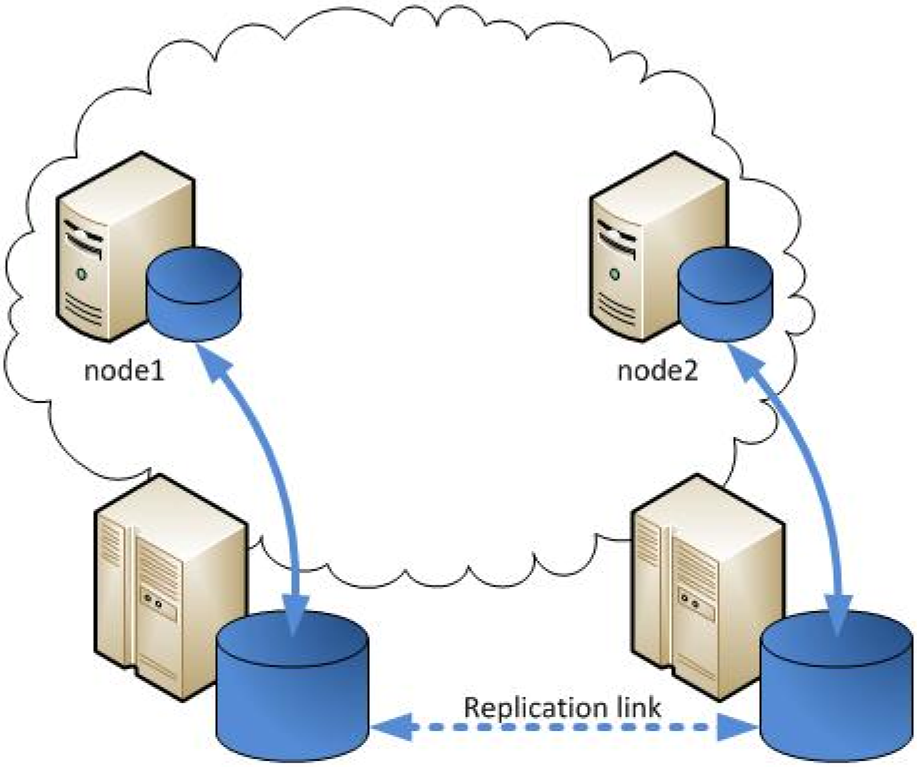
With this solution the logical storage space provided by the hypervisor is replicated through mechanisms residing directly in virtual guests. It can be used in active-passive cluster setups, as well as in active-active setups by using shareable file systems.

There are many products for delivering shared-nothing storage replication, and those products operate on different layers. There are block-based (DRBD, Veritas VR), filesystem based (GlusterFS) or even application-based (MySQL binary replication) products available, and each of them has its own advantages and disadvantages.

Because there is no logical interaction between such storage replication and the underlaying cloud layers, there is no explicit need for special support to those solutions, nor limitations or requirements to the underlaying physical storage or the methods in which the hypervisor provides this storage to virtual guests. Such setups are also highly scalable and can be fully managed by the customer in self-service.

By using zimory®cloud management stack, it is possible to automate the creation of such setups by using the "Plug" subsystem. This allows any user to create shared, highly available and cost neutral storage systems.

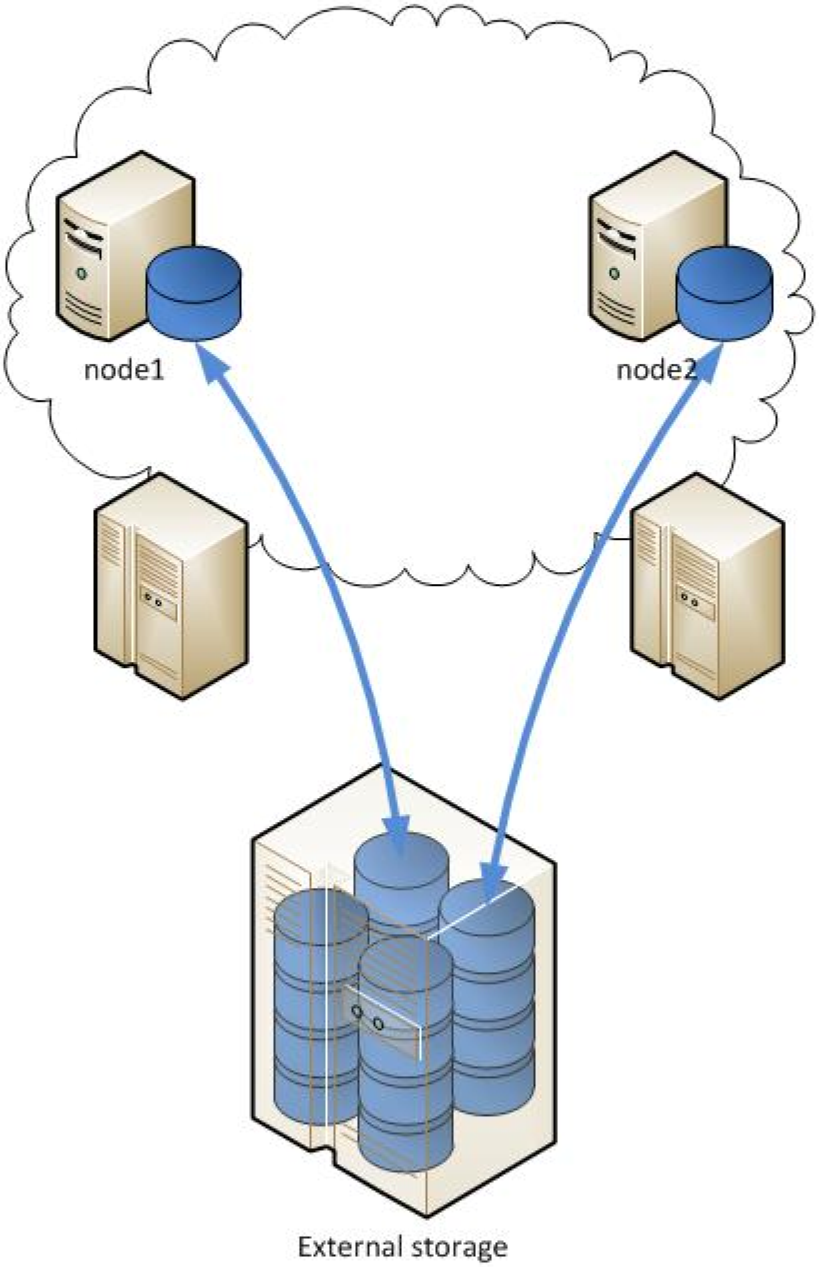
SHARED-NOTHING, HARDWARE-BASED STORAGE



Another method to provide shared-nothing storages is to bring the mirroring down to the hardware layer. Replication is done by the VHost operating system, or even the storage itself, and needs to be operated by the cloud provider.

The major issue with this kind of solutions is their missing flexibility in the day-to-day use. However, and because of their much better performance when compared to cloud-based shared storage solutions along with their attractive prices, they can be considered for small, private cloud scenarios.

SHARED, CLOUD-BASED STORAGE



This method constructs the best practice scenario for private cloud setups where high IO performance is needed for virtual machines. Logical volumes, provided by an external storage, are mapped by the hypervisor directly into virtual machines. Since there is only a very thin abstraction layer between the virtual machine and the storage, I/O bandwidth and latency is limited mostly to the connection stack (like iSCSI or Fibre Channel).

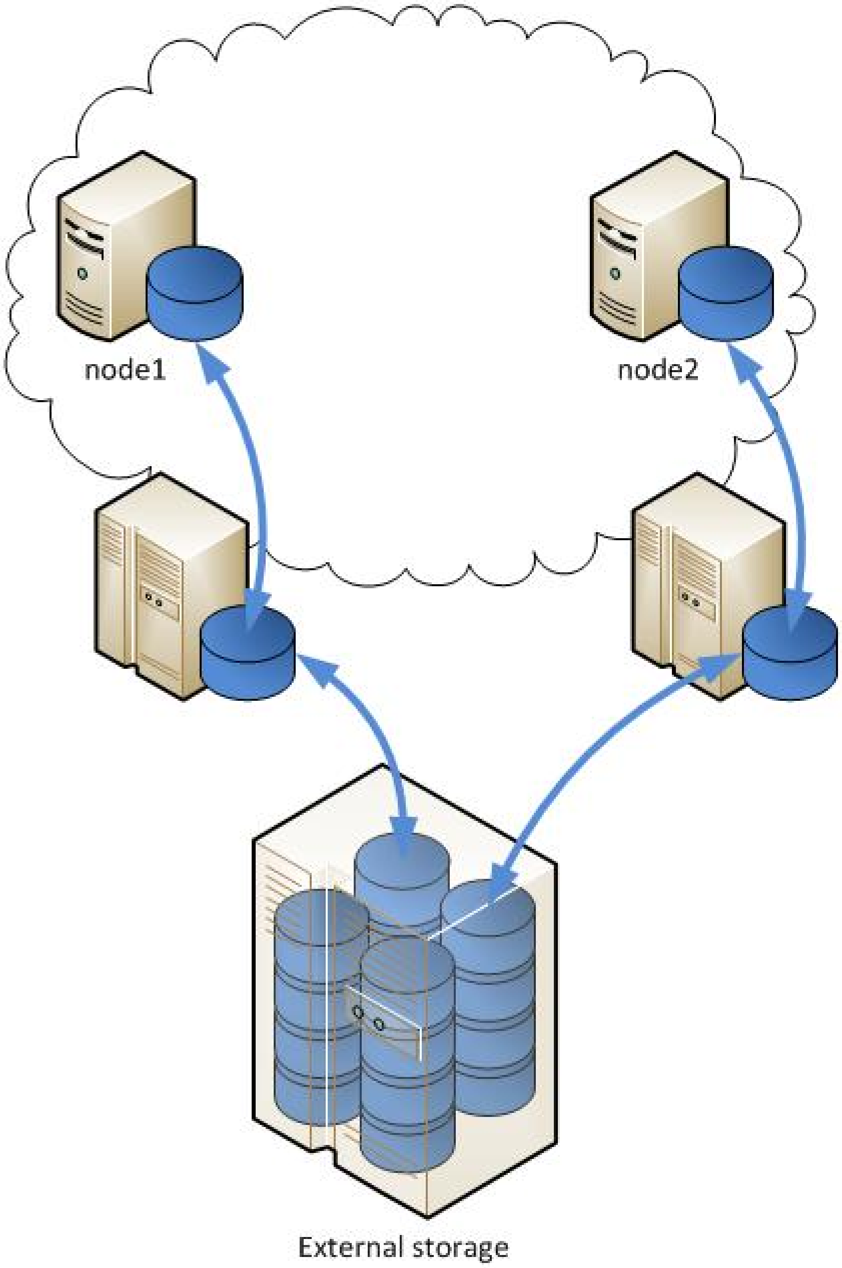
Again, this method can't be used to provide transparent, fault tolerant storage space. If mirroring is part of the storage system, mostly active-passive setups managed by the virtual guest operating system are possible. The mirroring itself should be done by the storage system. Explicit implementations have been performed by nearly every storage vendor.

zimory®cloud management stack does include the functionality to map block devices into virtual machines, although at the moment, this mechanism is implemented only for NetApp storages. For other vendors and technologies the cloud provider needs to map these devices manually and using possibilities and functions from the chosen hypervisor.

From the HA service delivery view, shared storage will always be a single point of failure. Even when mirroring, enterprise grade storage clusters can't provide write access to non-aware clients, so additional integration steps need to be considered by customers. Because of strong efforts on the storage vendor side, systems are currently able to provide very High Availability classes, which effectively facilitate problem resolution.

Once more, zimory®cloud is able to create mirroring-aware deployments by implementing the needed steps into its “Plug” subsystem. However, the cloud provider needs to create manually every single logical device exported by the storage system.

SHARED, HARDWARE-BASED STORAGE



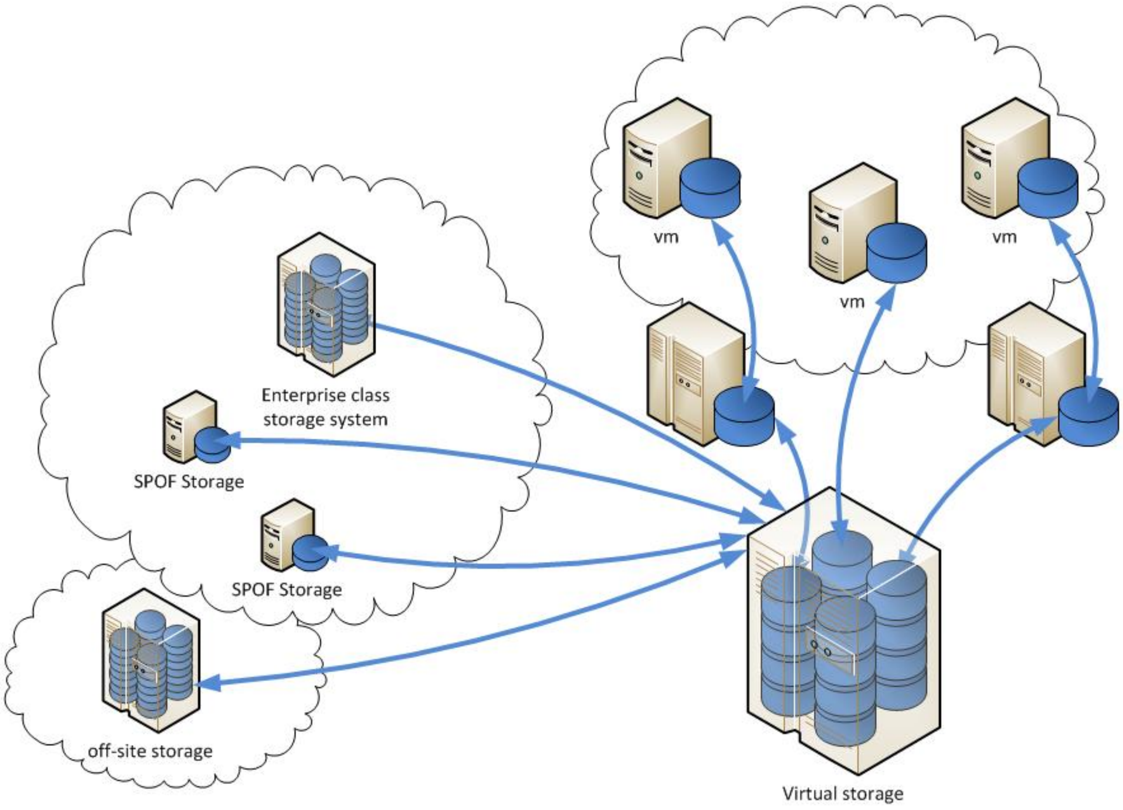
This is considered a best practice technique for big, public clouds where minimal administrative effort is targeted. The exported block devices from the storage are mapped into the VHosts and provided to the guest systems on a file system basis. This method is very similar to shared cloud-based storage solutions, but uses an additional abstraction layer between guests and storage, which emulates block devices writing to the hypervisors file system.

The central benefit of such setups is the high scalability without changes in the storage system itself. As long as there is enough space, many virtual guests can use the same physical storage without any further security considerations.

STORAGE VIRTUALIZATION BENEFITS FOR CLOUD COMPUTING

Storage virtualization is the new technology to support the cloud concept of the storage layer. The main purposes of storage virtualization is to consolidate different storage vendors, technologies and locations into one, easy to manage and always available logical storage node. This technology provides a sort of benefit to the storage availability layer, but also brings some basic problems with regards to single point of failure.

By using storage virtualization the cloud provider is able to significantly raise the availability class of his storage layer, when it comes to horizontal scalability, migration and mixed technologies. The other great benefit is the possibility to build a high available storage cluster based on standard components, which lowers the buy-in and ownership costs for high class HA storages.



ADDITIONAL CONSIDERATIONS ABOUT HIGH AVAILABLE BUSINESS PROCESS DELIVERY

Any technology has one important weakness: The human implementing and using the technology. To build highly reliable systems, not only technical, but also organizational precautions need to be considered. The most important are the following:

- It is absolutely vital to the availability of a system to create plans for handling problems and recovering from tough situations. Any thinkable scenario must be analyzed and optimal recovery strategies must be developed. Those plans must be trained, reviewed and improved continuously. Don't rely on improvisation capabilities in worst case scenarios.

- Every placed fail-over and fail-back procedure must be tested periodically. The fanciest technology is not worth a dollar if it does not work.
- Do not make changes to HA systems without having a strict change management process in place. Sometimes a quick fix to a small problem brings some other very large problems with it.
- Measurement of every possible key performance indicator is a very good entry to visualize how the used technologies and processes perform in real life. A close monitoring of every component needs to be implemented, and the generated data must be frequently reviewed. This is the only possibility to achieve optimal cost-performance ratios on your system.

GLOSSARY

HIGH AVAILABILITY TERMINOLOGY

Term	Definition
Cluster	A combination of two or more machines combined in a transparent manner. A cluster seems to be one logical machine.
Node	When referring to a single component of a cluster, the term node is used.
Hot standby cluster	Also failover or hot spare cluster. Clustering method, where one node of the cluster does the work (the primary node), while another monitors its operational status (secondary node). In case of a failure of the primary node, the secondary takes over the workload (nearly) instantly and becomes primary. A hot standby cluster with only one node left is referred to as being degraded.
Cold standby cluster	Also cold spare cluster. The cheapest method to do failover clustering, but the secondary node needs to be powered (or even ordered, delivered, staged and activated) in order to replace the former primary node.
Load balanced cluster	All cluster nodes are working simultaneously. In case of failure of one of the nodes, the additional workload is distributed among all other nodes. This clustering technique needs application support, such as session sharing, and sometimes concurrent access to the same data. It also needs a managing component such as a load balancer.
Fail-over	The process of replacing a failed node in a hot standby cluster. A switch-over is a controlled fail-over triggered not by a failure, but on purpose.
Fail-back	Recovery to normal operations after a fail-over process. Not needed if cluster is symmetric.
Synchronous data mirroring	When writing to mirrored storages, an IO-write request from the application needs to be physically written on any mirror, before it is acknowledged as done.

Asynchronous data mirroring	Any IO-write request is acknowledged as done to the application, as soon as it is written on one side of the mirrored storage space

STAKEHOLDER

Term	Definition
Provider, Cloud Provider	This is the organization or part of an organization, responsible for the underlaying hardware and software, such as host-OS and hypervisors. This role includes all combinations for any outsourcing or responsibility models (All provider levels). In other words, this is the one paying Zimory.
Customer	One or multiple users with access to the cloud management stack (zimory®manage). It can be, but must not be, related to the cloud provider (i.e. department). In other words, this is the one paying the cloud provider.
Consumer	Anybody using services from the application layer provider and from inside the virtual machines. Short: This one pays the customer.

VIRTUALIZATION

Term	Definition
vhost, virtualization host, host	This is the hardware running, any hypervisor, including the hypervisor itself and its underlaying OS (if any).
guest, vm, virtual guest, virtual machine	This is the virtual machine including the operating system running on virtual hardware provided by the hypervisor inside the cloud.

ZIMORY®CLOUD TERMINOLOGY

Term	Definition
zimory®central	This is the main management module. It is responsible for every interaction with the virtualization layer.
zimory®gateway	Provides network connectivity as well as network management for virtual guests.

zimory®manage	The user GUI. Every interaction between the customer and the cloud is done with help from this module.