

StarWind NFS

Introduction

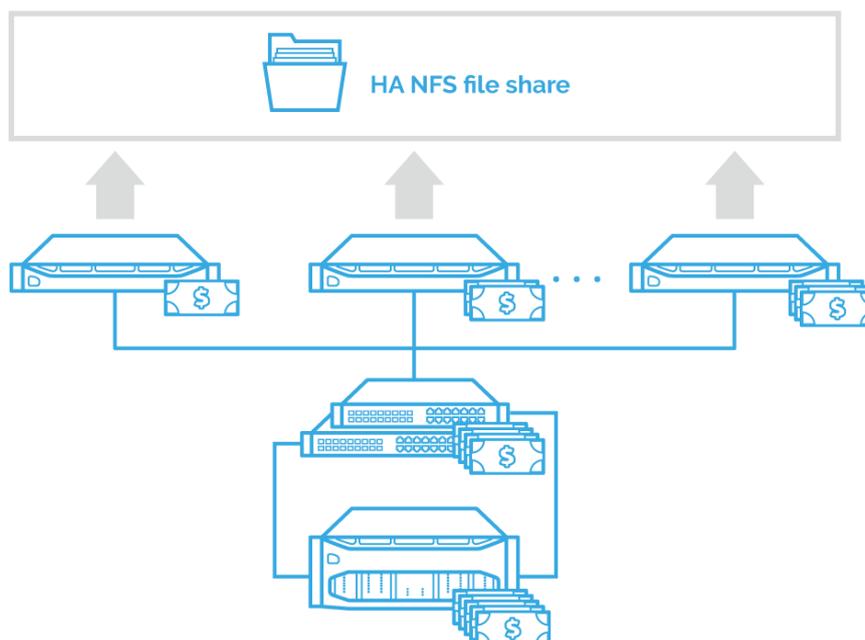
The Network File System (NFS) is a distributed file system protocol originally developed for allowing a client to access files as if they're locally available. Over the years, it has seen many version enhancements and can now be considered as a strong candidate for hosting enterprise workloads.

For the time being, NFS is easier to manage than FC or iSCSI for vSphere VM storage. Another widespread use is common file sharing.

NFS is mostly adept to simplify the infrastructure scalability, making it more popular than ever for providing file sharing access. However, traditional configurations of an HA NFS storage introduce some new challenges.

Problem

The main table-turner is the CAPEX of the implementation that can be explained by the requirement of shared storage in one form or another. More often the demand of NFS is met with the installation of additional servers + non-redundant JBODs or even costly SANs. Sometimes, the configuration gets even more complicated: gateway servers are added on top of older SANs to convert existing block storage to a more flexible NFS.

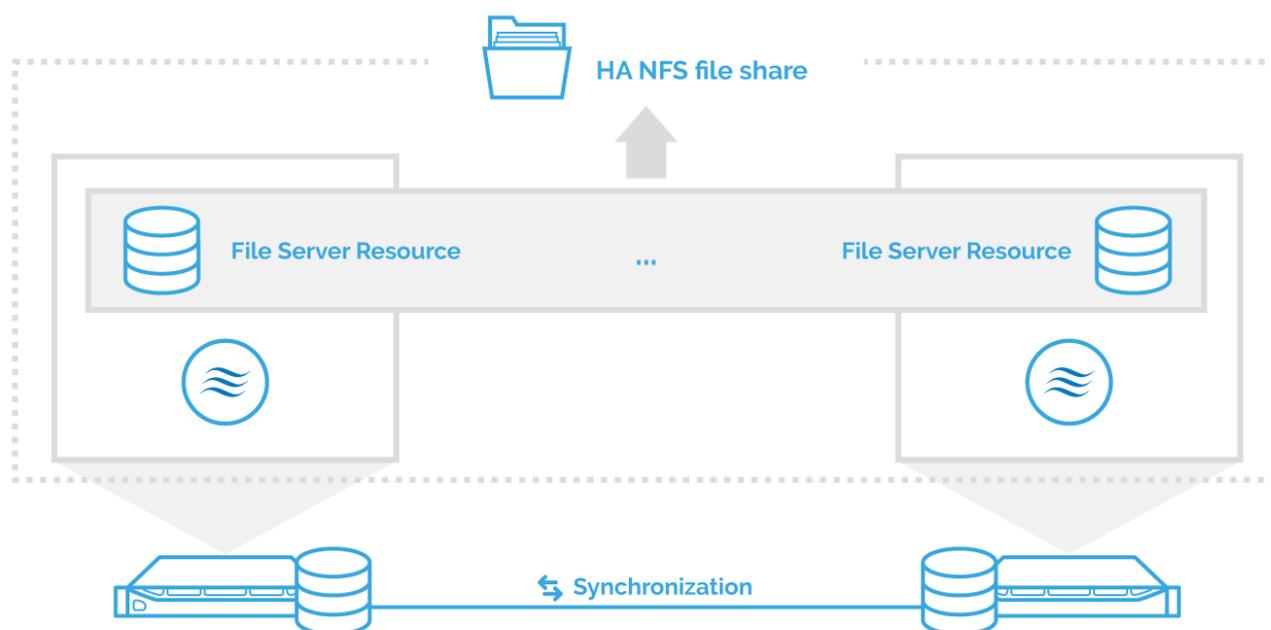


Something else to watch, is the low solution's ROI. With all of the available hardware taken into account, the performance improvement with all other additional necessities require considerable investments.

And last but not least – the low CLV. It is due to services available in existing solutions are unable to cover the growing demand of data services coming along with data consolidation. They are usually limited, so it becomes a tremendous challenge to get a unified redundant platform with a necessary set of data services like caching, snapshots, QoS, or data optimization features. For hardware solutions, adding new services may become another issue as vendors are more interested in selling the latest model, and not upgrading the existing ones.

Solution

The cornerstone of the improvements of CAPEX and OPEX is a shared nothing approach. StarWind shared nothing approach significantly shrinks both CAPEX and OPEX of the solution. Shared nothing means that StarWind minimizes the storage cluster footprint by eliminating the need for dedicated shared storage. Instead, it merges the storage and compute layers virtualized on commodity servers. Thanks to this approach, StarWind can minimize the cluster footprint to as low as 2 nodes.



The ROI of the solution is also improved. It's due to the multilevel redundancy and high-performance architecture of StarWind. To improve the performance, StarWind takes the advantage of DRAM and Flash caching functionality. The unnecessary CPU overhead is minimized by implementing RDMA for data synchronization.

Finally, StarWind delivers excellent CLV by providing a rich feature set that includes caching, snapshots, QoS, and data optimization features. Moreover, this list can be extended on the fly with a simple software update throughout the entire storage solution lifecycle.

Conclusion

StarWind enhances the capabilities of NFS protocol, providing high performance and high availability for mission-critical applications.

It improves CAPEX and ROI by maximizing the hardware utilization and introduces an excellent CLV by providing data services that were only previously available in enterprise-level storage solutions.