WHITE PAPER

pernixdata

PernixData FVP™

Technical White Paper

EXECUTIVE SUMMARY

The last decade has seen virtualization become a mainstay in the enterprise data center. Enterprises are now looking to virtualize their most performance sensitive applications. Trends in hardware such as multi-core processors are allowing customers to run even more virtual machines per physical server. The result has been a tremendous increase in the demand for storage performance in virtualized environments. Yet storage technologies have failed to keep up.

Virtualized environments are deployed with network based, shared storage architectures. These storage technologies have made great strides in providing high quality capacity and data services. Yet the performance gains they have delivered to customers have been mediocre at best. Enterprises have thus been forced to over-provision storage to get anywhere near the performance they need for their virtualized environments - an expensive and operationally disruptive proposition. It is no surprise then that storage is the number one expense that most enterprises incur in their virtualized data centers.

Storage is therefore the single largest impediment for the further adoption of virtualization within the enterprise. In this whitepaper we will delve into how enterprises can use PernixData FVP™ to overcome these challenges and fundamentally rethink the design of the virtualized data center.

Introduction

If leveraged correctly, server-side flash can herald a new era in virtualized data center design. This whitepaper introduces PernixData FVP - an enterprise class, scale-out, software tier, based on server-side flash that is specifically catered to satisfying the performance needs of the virtualized data center.

Audience

This whitepaper is intended for PernixData customers and partners who are considering using PernixData FVP in virtualized data centers. This whitepaper assumes a basic understanding of virtualization and flash technologies.

Technology

- **Flash Devices**: Server-side PCIe or SSD storage devices.
- **Flash Cluster**: Cluster-wide pools of flash devices created by PernixData FVP.
- **SAN**: Network-based shared storage typically used in virtualized data centers.

Virtualization and Flash Storage - An Ideal Combination

SANs satisfy two disparate requirements in today's virtualized data centers. Firstly, SANs satisfy the need for capacity management and data services such as snapshots and backups. These data services are critical for data protection in disaster recovery situations. Secondly, SANs satisfy the performance requirements of virtual machines (VMs) running in the data center.

While SANs are an ideal solution for Capacity & Data Services, they have failed miserably to satisfy Performance. This is because SANs are designed with mechanical drives in mind. Unfortunately mechanical drives have evolved very little in the performance realm.

Flash storage, because of its performance characteristics, promises to be the ideal solution for a virtualized data center's 'Data in motion' requirements. But how does one best leverage flash in order to satisfy performance requirements? Below we discuss what the appropriate location for flash storage is within the virtualized data center.

Appropriate Location for Flash Storage - In the Server

Flash can be located either within a SAN or it can be located on the server itself. Let us evaluate each approach in detail to identify the appropriate location for flash storage.

When customers procure a SAN with flash inside, it usually requires a rip and replace of their existing storage resulting in huge disruptions. Note that the existing SAN is usually doing a good job satisfying the Capacity & Data Services needs. Yet, to meet the Performance requirement, we have to purchase a new SAN, which includes additional Capacity & Data Services. Clearly this is inefficient.

Even if we assume that we can somehow introduce flash inside the SAN without a rip and replace of existing storage one has to wonder whether flash in the SAN is able to satisfy Performance needs at all. Accessing flash inside a SAN involves going over the network. Network latencies are usually orders of magnitude more than flash latencies. This means that once a flash array is introduced the network becomes the new bottleneck and will completely obfuscate any gains that flash provides. Finally, flash inside a SAN usually sits behind a legacy disk controller - one that was designed with mechanical drives in mind. This too impacts how well the flash within a SAN can be leveraged. It is then fair to claim that incorporating flash within a SAN is akin to driving a sports car in a busy city street. One can, therefore, safely conclude that flash within the SAN as a solution for Performance is simply dead on arrival. The correct location for flash is then on the server-side where VMs can leverage it for Performance needs without incurring any network latencies.

PernixData FVP - An Enterprise Class, Server-Side Performance Tier

PernixData FVP is the only enterprise-class, software-defined Performance tier for server-side flash. We claim thisfor a number of reasons. These include:

1) Scale-out architecture
2) Read and write acceleration
3) Transparent support for clustered hypervisor functionality
4) Operational ease of use

Scale Out Architecture

PernixData FVP is architected with a scale-out design philosophy. In a typical virtualized data center hosts, containing server side flash, can be introduced on demand. PernixData FVP seamlessly leverages these new resources and performance scales commensurately. This is in contrast to a typical SAN that cannot scale beyond a certain number of hosts.

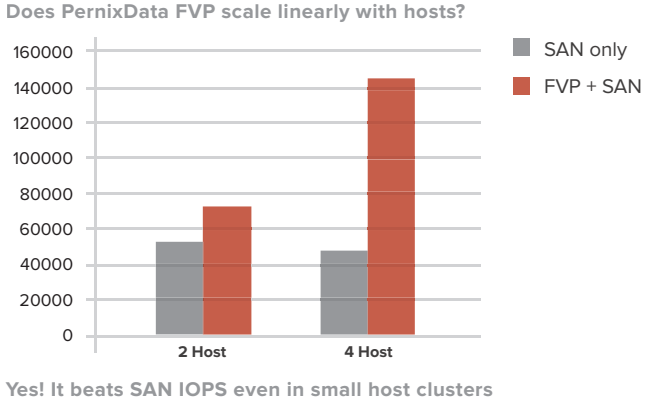

Figure 1 *shows how PernixData FVP's scale-out architecture makes it an ideal Performance tier. The contributions of both SAN and PernixData FVP to the total IOPS were also measured. Firstly, we observe that the SAN in absolute terms does not scale as the number of hosts increase. Secondly, we notice that even though the SAN does not scale the total IOPS scales with the number of hosts. This is because PernixData FVP, acting as the performance tier, is able to scale and leverage the new hosts and their flash devices as the environment grows. Results similar to those delivered by PernixData FVP running on 2-4 hosts cannot be obtained by simply buying a bigger SAN. This is because a best in class SAN cannot deliver the absolute number of IOPS that PernixData FVP is able to deliver using 2-4 hosts.*

Read and Write Acceleration

The idea of servicing reads from the server side flash is straightforward. Simply use the flash as a read cache. What about writes? Let's say you edit data sitting on the server side flash and it never makes it to the underlying SAN. This could happen because, say, a server went down. In other words, you lose data. This is a huge problem. Enterprise class platforms based on server side flash must be able to accelerate both reads and writes equally well while providing appropriate failure handling. PernixData FVP does exactly that.

PernixData FVP allows VMs to run in one of two modes:

1) **Write Through**: In this mode writes issued by a VM are acknowledged only when they are successfully committed to both the server-side flash and the underlying SAN. Write Through mode is useful for workloads that have a good mix of read and write operations. In Write Through mode all read operations are satisfied from the server-side flash device whereas write operations are executed both against the server-side flash and the backend SAN in a 'pass-through' fashion. The write is acknowledged only when writes to both devices succeed. Data consistency is therefore guaranteed because the underlying SAN is the 'data of record.' An interesting observation about Write Through mode is that write performance increases even though they are happening synchronously against the SAN. This is because PernixData FVP completely offloads all reads from the SAN and so the SAN is left servicing only writes thereby enhancing performance considerably.

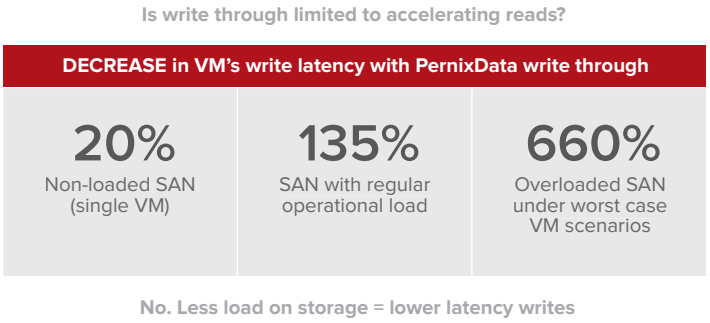

Figure 2 *illustrates how Write Through mode benefits not only read operations but also write operations as a side effect.*

2) **Write Back**: In this mode writes issued by a VM are acknowledged when they are successfully committed to the server-side flash. Data is then de-staged from the server-side flash to the SAN over a period of time asynchronously Write Back is useful for VMs that have low latency requirements for write operations. This is because these writes now incur the flash device latencies as opposed SAN latencies. One of the concerns that come to mind when evaluating Write Back mode is that of failure handling. PernixData FVP overcomes the failure handling issue for Write Back VMs by replicating data across the cluster. In the case of a failure PernixData FVP simply fetches the data from the replica on another host. The need for failure handling however varies from one use case to another. For example, production OLTP databases have very stringent needs for failure handling whereas VDI VMs or test and dev VMs do not. As a result PernixData FVP allows users to configure the number of replicas when a VM is run in WriteBack mode. A user can choose among 0, 1 or 2 replicas for the data.

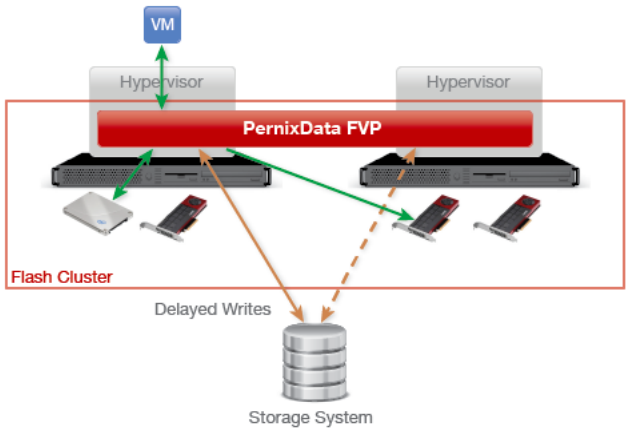

Figure 3 *illustrates how data is replicated across the cluster when a VM is configured in Write Back mode. The green arrows indicate how when the VM issues a write operation data it is both written to the local server-side flash and replicated to remote hosts.*

Transparent Support for Clustered Hypervisor Functionality

One key requirement for any solution that is employed in the virtualized data center is transparent support for typical hypervisor features. For example, VMware™ vMotion™ is a feature that transparently migrates VMs from one physical host to another. Many times this occurs unbeknownst to the user. A Performance tier must be able to continue to support such features without requiring any changes to the environment or any changes to existing operational workflows.

PernixData FVP is built to support clustered hypervisor features, such VMware's vMotion and DRS, transparently. When a VM moves from one host to another it does not lose its footprint on the server-side flash. Instead, it simply remotely accesses the server-side flash on the host it was running on previously and populates its now new local server-side flash with this data. In other words, a VMware vMotion or DRS does not cause the virtual machine to drop its server-side flash footprint. As a result, a VM while migrating sees very little impact on performance.

In addition, PernixData FVP does not require new operational workflows or manual intervention in order for VMware vMotion or DRS to work. Because of its transparent nature PernixData FVP guarantees that clustered hypervisor features operate no differently when PernixData FVP is in the environment.

Operational Ease of Use

It is critically important that an enterprise class platform be easy to deploy, operate and manage. PernixData FVP has focused heavily in making sure that it creates zero disruption to an enterprise's existing investments and workflows. PernixData FVP embeds itself completely transparently within the hypervisor. There is no need for guest agents or virtual appliances both of which introduce tremendous inefficiencies and single points of failure. PernixData FVP also does not require static partitioning of flash or configuring flash sizes per VM or VMDK. In addition, users do not need to change their operational workflows in order to leverage standard hypervisor functionality such as HA, DRS or vMotion. Finally, PernixData FVP is agnostic to server, storage or flash models and works equally well in any setup. PernixData FVP also supports heterogeneous environments seamlessly.

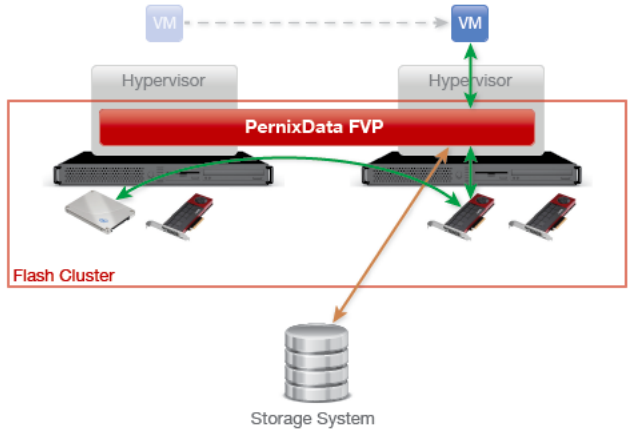

Figure 4 *shows how when a VM migrates from one host to another it does not drop its server-side flash footprint for its read and write operations. Instead it simply goes over the backplane to access data on the remote flash (green arrow). This is usually much faster than accessing SAN for the data.*

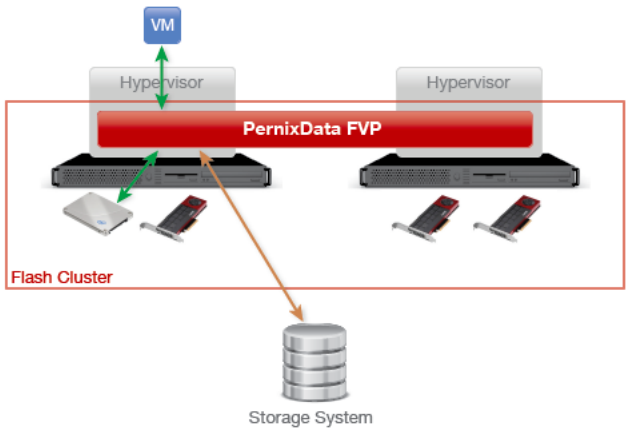

Figure 5 *illustrates how PernixData FVP embeds itself transparently inside the hypervisor. In addition, Figure 5 also shows how PernixData FVP operates in heterogeneous environments. For example, the host on the right has 2 PCIe cards, whereas the host on the left has one PCIe card and one SSD card.*

CONCLUSION

Storage is the single largest impediment for the penetration of virtualization within the enterprise data center. Existing network attached, shared storage is unable to keep up with the performance requirements of virtualization because of its antiquated design based on mechanical drives. However, these storage systems do a good job satisfying the capacity and data services requirements in the data center.

The advent of flash storage holds promise for satisfying the performance requirements provided it is used appropriately. Simply adding an all flash array inside a SAN is not the right answer because it is both operationally disruptive and simply moves the bottleneck to the network.

PernixData believes the correct location for flash in the data center is on the server side. However, existing solutions for leveraging server-side flash are not enterprise class. Instead they are caching solutions that satisfy niche use cases and cause operational disruption. Because of its scale out architecture and built-in clustering capabilities PernixData FVP is able to accelerate both read and write operations while guaranteeing no data loss. PernixData FVP supports clustered hypervisor features, such as VMware vMotion and DRS, transparently and without the need for special workflows or proprietary interfaces. Finally, PernixData FVP embeds itself inside the hypervisor transparently and with zero disruption. This means enterprises can install PernixData FVP within minutes and reap immediate return on investments.

As a result PernixData FVP is the first truly seamless and enterprise class, software-defined, scale-out Performance tier that leverages server-side flash for virtualized data centers.