

# **Lab Validation: Optimizing Storage for XenDesktop with XenServer IntelliCache**

**Reducing IO to Reduce Storage Costs**

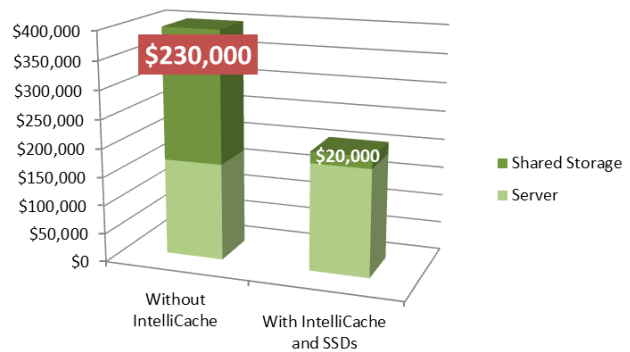


# Table of Contents

|     |  |    |
|-----|--|----|
| 1.  | Introduction .....                         | 2  |
| 2.  | Executive Summary.....                     | 3  |
| 3.  | Reducing IOPS Minimizes Storage Costs..... | 4  |
| 4.  | What is IntelliCache?.....                 | 5  |
| 5.  | Test Results and IOPS Demand .....         | 9  |
| 6.  | Best-Practice Recommendations.....         | 16 |
| 7.  | Conclusion.....                            | 18 |
| 8.  | Appendix A: Testing and Configuration..... | 20 |
| 9.  | Appendix B: Test Infrastructure .....      | 21 |
| 10. | Appendix C: Enabling IntelliCache .....    | 26 |

## 1. Introduction

One of the major barriers to the adoption of virtual desktop infrastructure is the high cost of the required shared storage. Within virtual desktop environments, high IO latency and resulting bottlenecks negatively impact the user experience. Consequently, sizing storage infrastructure with sufficient IOPS is crucial. However, the price of storage rises as IOPS capacity increases, which can quickly erode the value proposition for virtual desktop infrastructure.



*For a 1000 desktop VM deployment, IntelliCache can potentially save \$210,000 in shared storage costs.*

Virtualizing your XenDesktop deployment on XenServer and enabling the XenServer IntelliCache feature decreases shared storage costs. IntelliCache reduces IOPS requirements by caching boot images and non-persistent or temporary data on the local XenServer host. This caching decreases IOPS on shared storage arrays, potentially saving thousands of dollars. In a 1000 desktop deployment, Citrix estimates the XenServer IntelliCache feature could result in cost savings up to \$210,000.<sup>1</sup>

XenServer Performance Engineering testing revealed the following:

- In our environment, IntelliCache resulted in a 92% decrease in IOPS on shared storage.<sup>2</sup>
- Pooled Desktop configurations provide the biggest opportunity to save costs because they reduce both Reads and Writes on shared storage. However, Dedicated Desktop configurations still benefit from the reduction in Read IOPS on shared storage.

The goal of this paper is to outline how IntelliCache works as well as provide data and example scenarios. This paper presents a configuration of XenDesktop with IntelliCache enabled. XenDesktop is configured to use Machine Creation Services (MCS) and Pooled Desktops (also known as *shared desktops*).

This paper demonstrates the performance improvements that come from enabling IntelliCache and provides data on the reduction in load (IOPS) on shared storage. Furthermore, this paper explains how you might achieve similar benefits by using a few different scenarios to show how to establish a baseline, enable IntelliCache, and observe the decrease in IOPS.

Like always, it is important to test IntelliCache in your own environment before ordering storage and not to rely merely on the results in this paper since your results may vary.

<sup>1</sup> One thousand shared virtual desktops configured with 1.5 GB of memory on blade servers and medium workloads could save you approximately \$210,000. The prices are just for guidance; seek pricing for your needs from reseller. To get an idea of your own potential savings, see the XenDesktop/XenServer Deployment Cost Calculator at <http://www.citrix.com/xenserver/features/advanced-integration/intellicache-savings>.

<sup>2</sup> In practice, the degree to which IntelliCache decreases IOPS for shared storage in your environment depends on a variety of factors, including the number of hosts and number of VMs per host.

## 2. Executive Summary

IntelliCache delivers on its promise to reduce IO load and shared storage costs. You can potentially reduce shared storage IOPS requirements by over 92%—and in some phases of the lifecycle over 99%—using IntelliCache. It is important to note that the use of IntelliCache does not decrease the IOPS; it simply redirects the IO operations to less costly local storage.

*IntelliCache* is a XenServer feature that caches temporary and non-persistent operating-system data on the local XenServer host. When IntelliCache is enabled, a portion of the virtual-machine runtime reads and writes occur on low-cost local storage.

**Read Cache.** The local storage location on the XenServer host where operating system data is stored when IntelliCache is enabled.

**Write Cache.** The local storage location on the XenServer host where desktop virtual machine data is stored when IntelliCache is enabled.

**Cold Cache.** When IntelliCache is enabled and the cache has yet to be fully populated, it is known as a *cold cache*.

**Warm Cache.** When IntelliCache is enabled and the cache is largely populated and the number of reads to the master image decrease, it is known as a *warm cache*.

**Login VSI.** Login VSI is a benchmarking tool that lets you measure the performance of centralized desktop environments by simulating user workloads, such as Microsoft Office.

**Machine Creation Services (MCS).** MCS is a XenDesktop provisioning mechanism that provisions, manages, and decommissions hosted desktops through hypervisor APIs (XenServer, Hyper-V, and vSphere). MCS lets several types of VMs be managed in a catalog in Desktop Studio, including dedicated and pooled machines.

During our testing in the XenServer Performance Labs, enabling IntelliCache ultimately led to a decrease on the shared storage from 1378 IOPS to 2.1 IOPS. IntelliCache achieved these savings by moving the reads and writes to occur on local storage instead of shared storage, which reduces the VM's need to read from and write to shared storage.

Testing reveals three notable measurements virtualization architects need to consider:

1. The IOPS when the first user logs on to his or her desktop while the operating system data is cached in the Read Cache on the local hard drive (known as *User Log On/Cold Cache*).

2. The IOPS, as users log on, after the Read Cache is populated (known as *User Log On/Warm Cache*).

3. For environments that do not want to use the XenDesktop hypervisor throttling and power management features, the number of IOPS that occur during boot with IntelliCache (known as *Cold Cache*).

It should be noted that our testing was not designed to provide virtual-machine density numbers but rather to demonstrate the ability of IntelliCache to reduce IOPS on shared storage.

**92%**  
**Reduction  
in IOPS  
on Shared  
Storage**

Based on testing a simple example of 90 VMs, when IntelliCache is enabled, most IOPS (under 550) occur when the desktop VMs boot. While some configurations might want to factor in boot IOPS, XenDesktop hypervisor throttling and power management features can mitigate this impact.

In our baseline Login VSI test run, monitoring from the NetApp storage, we saw that with 90 VMs, the number of IOPS peaked around 1378.

We then enabled IntelliCache and ran the same 90 user test and observed a 92% decrease in IOPS on the NetApp. This was due to IntelliCache caching data in the local server storage, thus redirecting the IO to local storage.

To keep up with the local IOPS demand, we used 2x Solid State Drives (SSDs) in a RAID 0 configuration. While using SSDs with IntelliCache is a best practice recommendation—SSDs can handle a far greater number of IOPS compared to traditional SAS/SATA drives—SAS drives can also offset IOPS on shared storage.

To further reduce IOPS demand, we followed the best-practice recommendation of using a RAID controller with Battery Backed Write Cache. However, while we subsequently determined this was not necessary for SSDs, we do recommend Battery Backed Write Cache for SAS drives.

In our testing, we used the following configuration:

- XenServer 6.0.2 and XenDesktop 5.6.
- An IBM x3650M3 with 2x Intel Xeon x5670 CPUs and 144GB of RAM running XenServer 6.0.2 to host 90 Windows 7 desktops.<sup>3</sup> Each Windows 7 VM was allocated 1 vCPU and 1.5GB of RAM. Our virtual disks were hosted on an NFS share on a NetApp (FAS3270) consisting of 17 spindles.
- For the workload, we used the Login Consultants Virtual Session Indexer (Login VSI) 3.0 medium workload. We used Login VSI to simulate a medium user workload in the XenDesktop virtual desktop environment.

### 3. Reducing IOPS Minimizes Storage Costs

Often, the first thing that enters many people's mind when evaluating storage for a deployment is *how much space do I need?* However, space requirements are only half of the consideration. The other equally, if not more important, consideration is *what are my IOPS requirements?*

While running out of space is problematic, failing to foresee your IOPS requirements can create bottlenecks, which result in an unacceptable user experience, or worse, an overall failure in your XenDesktop deployment. Likewise, your deployment can end up with inadequate scalability and density.

Storage is one of the most expensive and difficult to implement pieces of a virtual desktop infrastructure. A key factor in storage prices is the IOPS capability. Reducing shared storage IOPS requirements can potentially save significant amounts of money on your storage.

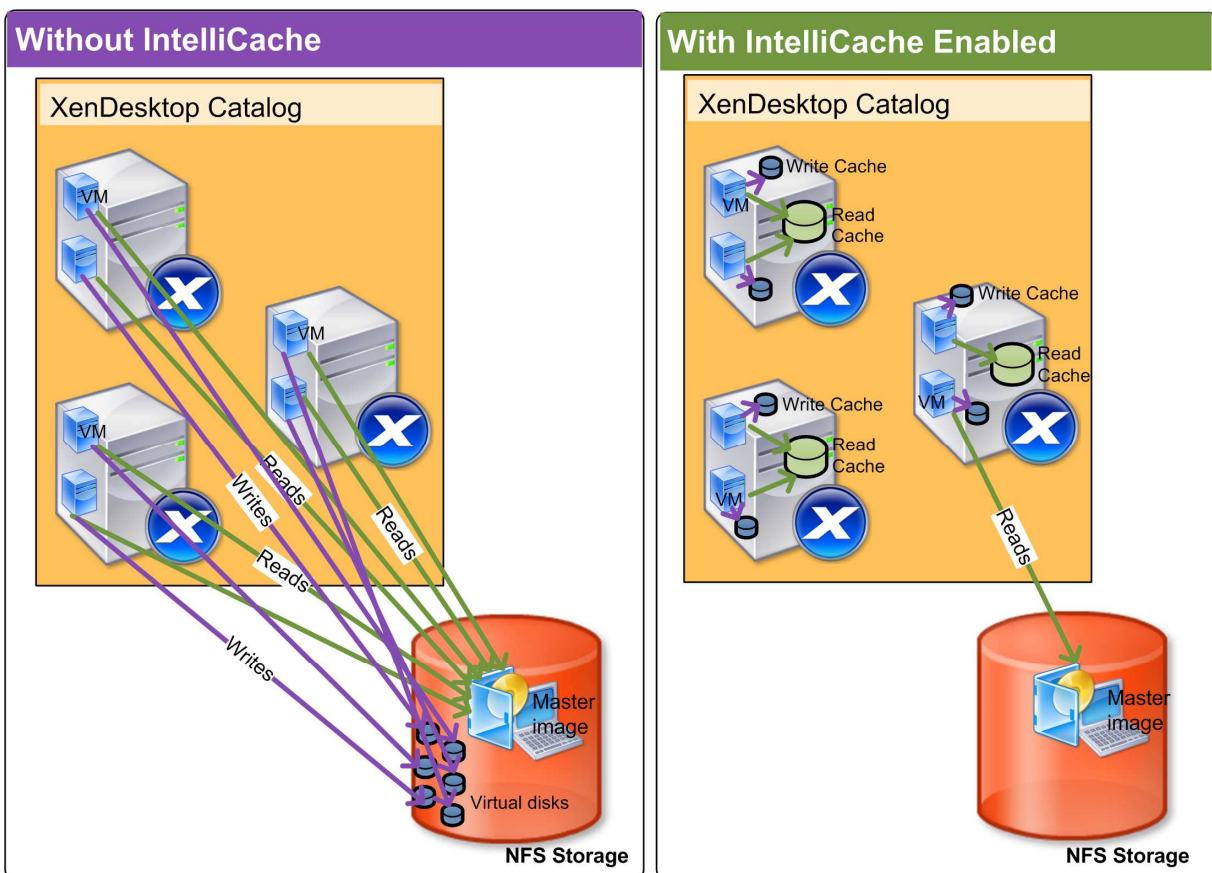
<sup>3</sup> We used ninety VMs on one host as an example. However, you can expect higher density than 90 VMs on a host as described in CTX131047—*XenServer 6.0 Configuration Limits*

## 4. What is IntelliCache?

IntelliCache is a XenServer feature that can be used in a XenDesktop deployment to cache temporary and non-persistent operating-system data on the local XenServer host. IntelliCache is available for Machine Creation Services (MCS)-based desktop workloads that use NFS storage.

In a typical XenDesktop configuration (without IntelliCache), desktop VMs read the operating-system data from a master image on a costly shared storage array. When IntelliCache is enabled, a portion of the virtual-machine runtime reads and writes occur on low-cost local storage: XenServer caches the operating-system files on its local hard drive in a Read Cache.

Likewise, when IntelliCache is enabled, each desktop VM writes to its own Write Cache on the local host, preventing writes to shared storage. As a result of caching on local storage, when IntelliCache is configured for a pooled desktop, it significantly reduces the load on the remote storage and the amount of network traffic. This is shown in the following illustration.



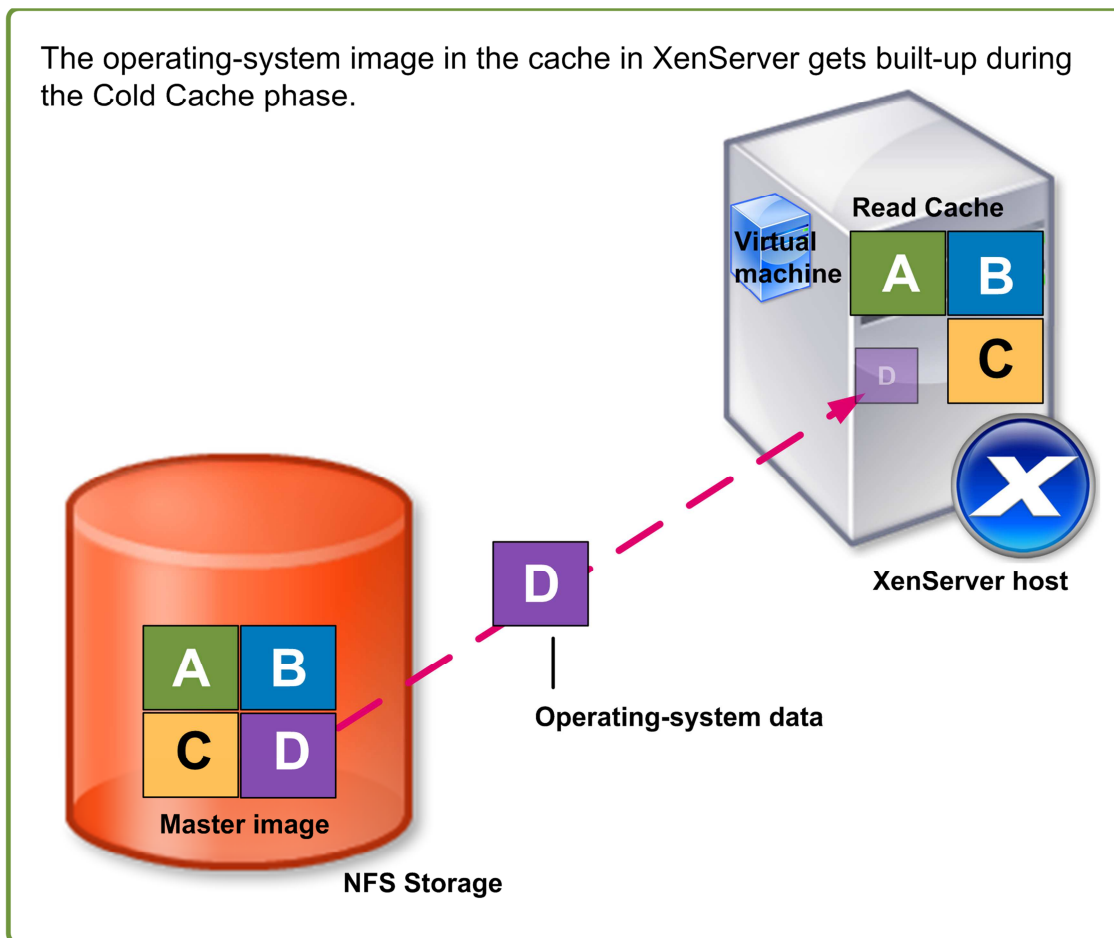
*Without IntelliCache, each desktop VM reads data from the master image on the shared storage and writes data to its virtual disk on the shared storage. However, with IntelliCache enabled for Pooled Desktops, desktop VMs cache most read data locally, so they only need to read from the shared storage when data is not available in their local cache. Likewise, desktop VMs write to their own Write Cache on local storage.*

## 4.1. The IntelliCache Caching Process

The VMs cannot benefit from the Read Cache immediately since it is not fully populated. Instead, XenServer populates the Read Cache progressively each time a desktop VM requests a specific block of operating-system data.

When the first desktop VM is powered on and XenServer creates the Read Cache in the local SR, the cache is empty and needs to be filled. A XenServer host caches blocks of the master image in its Read Cache each time its desktop VMs read data from the master image. When subsequent desktop VMs boot, they will read the already cached blocks and will not need to access the data from shared storage.

The illustration that follows shows how XenServer populates Read Cache as it reads the Master Image.

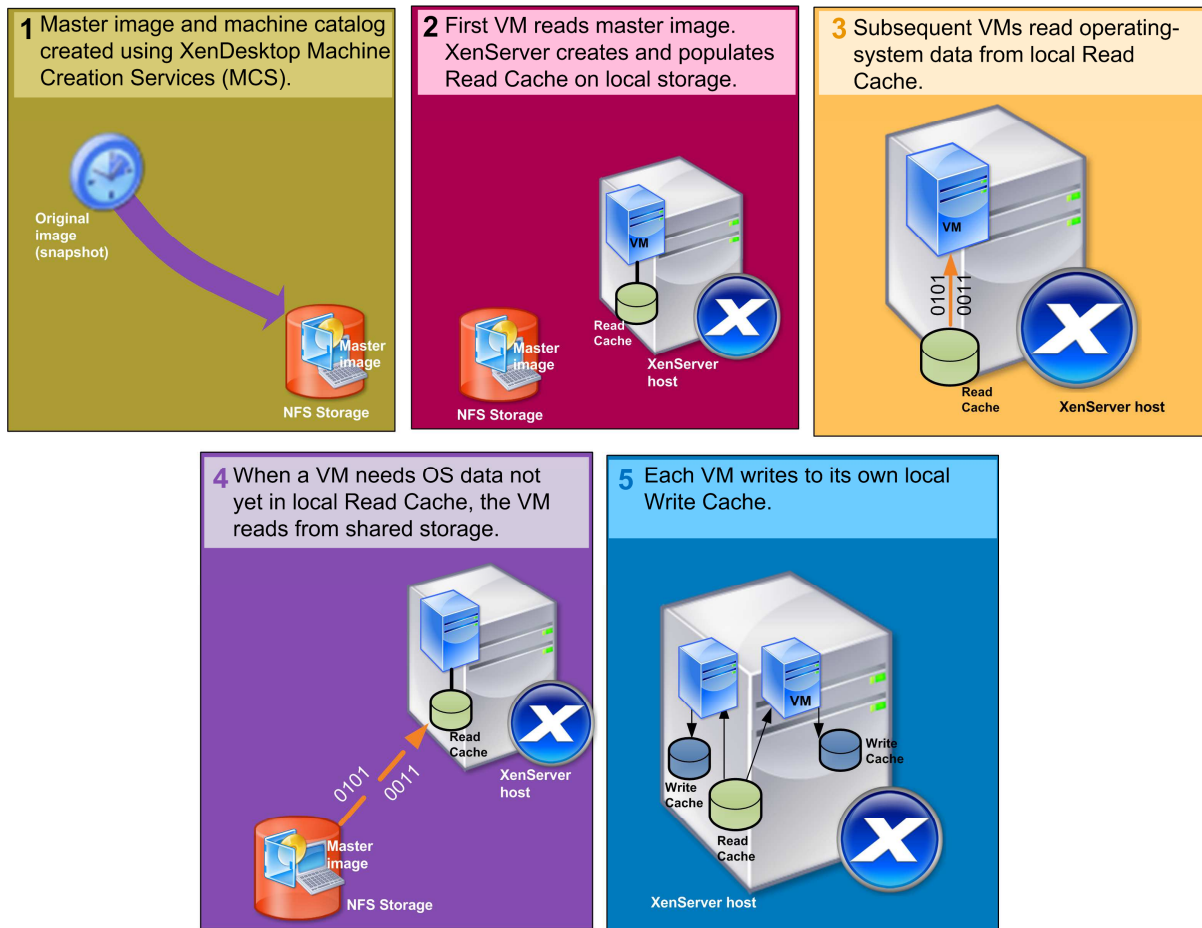


*This illustration shows how, when a desktop VM cannot find part of its operating system in the Read Cache on the local host, the desktop VM accesses the master image on the storage. As the master image is read, the Read Cache is populated with part of the missing image. In this illustration, “D” represents a block of data not found in the Read Cache.*

Each read of the master image reduces the number of times the desktop VMs in that catalog and on that host need to access the master image on shared storage. As the master image is read and more of the cache is populated, it decreases the IOPS demand on shared storage.



The following illustration shows the overall process when desktop VMs read from and write to local caches instead of shared storage.



*This illustration shows how IOPS are reduced, for Pooled Desktops, when VMs in the same machine catalog use IntelliCache. As shown in the last panel, all of the VMs read from the Read Cache.*

It should be noted that each catalog on a host results in another local read cache. When administrators update catalogs (for example, if there is an operating-system update released), depending on the rollout strategy, a user could still be running the older master image until that user reboots. Consequently, when you are sizing, keep in mind, there could be a period during upgrades when both the new and old catalog is used. It is important to remember that each active version of the catalog, including ones run simultaneously during updates, creates another local Read Cache.

## 4.2. Understanding When IOPS Decrease

While IntelliCache reduces write IOPS on shared storage immediately, read IOPS decrease over time while IntelliCache builds its Read Cache. Consequently, for the purposes of this paper, we draw a distinction between the stages of caching: when the Read Cache is being built and when it contains the majority of the master image in use.



When the cache has yet to be fully populated, we refer to it as a *cold cache*. When the VMs are rebooted or shut down and restarted, the Write Cache files are discarded; however, the Read Cache persists and still contains the cached data. As a result, the VMs can read from the Read Cache even after a reboot.

Since the Read Cache files can persist after reboot, the cached parts of the operating system can continue to accrue until no new parts of the image are requested. Once the Read Cache is largely populated, we refer to it as a *warm cache*.

It is when the Read Cache is in a warm cache state that we see the maximum reduction in IOPS on the shared storage. The VMs can obtain all of their operating system data from the Read Cache on the local hard drive—VMs no longer need to access the master image on shared storage.

During our testing, we observed two stages for each type of cache:

- **Boot Stage.** During the boot stage, we started 90 desktop VMs using the XenDesktop throttling and power-management features so the VMs started in a staggered manner.
  - **Boot Cold Cache.** When the first Desktop VM on a host boots for the first time, operating-system data the desktop VMs needs to start is stored in the cold Read Cache.
  - **Boot Warm Cache.** After XenServer uses data from the first VM to populate the Read Cache, VMs only need to read from the master image when they cannot find data in the Read Cache. It is during this stage of the boot process that we see IOPS significantly decrease and level off.
- **Login VSI Test Run.** For the Login VSI test run, we used 90 desktop VMs that were all booted. Every 30 seconds Login VSI launched a user to login. Once the user had logged in, the Login VSI medium workload is started. After all the users logged in and ran a workload, each user finished the test run and logged off.
  - **Log On Cold Cache.** When the first user logs on to a desktop, the desktop VM will require more operating-system data from the master image on the shared storage. Like the Boot Cold Cache stage, XenServer stores the data read from the shared storage in the local Read Cache.
  - **Log On Warm Cache.** After XenServer populates the Read Cache with log-on data, the desktop VMs can obtain most of their data from the local Read Cache.

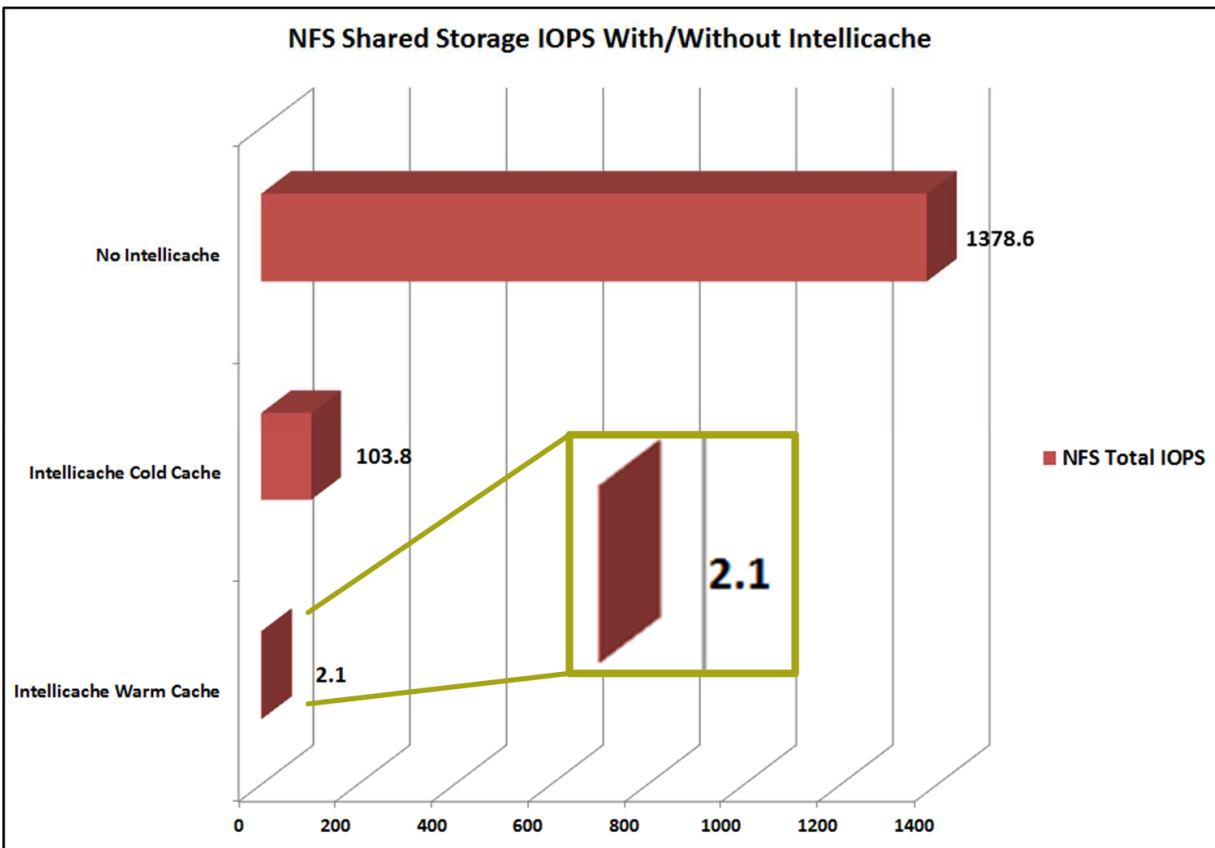
It should be noted that the terms *first desktop VM* and *first user* refer to the first VM or user on *each* host. Because the Read Cache is specific to a host, the Read Cache is built as soon as a VM on the host starts (the first VM ever to boot) or when the very first user that is connecting to host connects for the first time.

IntelliCache returns to cold cache mode whenever you apply an update or change the master image. When the master VM is updated, a new Read Cache is created, and the cache will be cold upon initial boot of the first VM. Therefore, when sizing shared storage with IntelliCache, we recommend you look at the IOPS requirements for when your VMs will be running in both cold- and warm-cache modes.

## 5. Test Results and IOPS Demand

In a typical VM lifecycle, VMs are booted, users log on to the desktop VMs, users perform their work, desktops may be idle for a period of time (for example, if users take a break), users logoff, and the desktop VMs may be offline. As a result, the most IO-intensive phases of the VM lifecycle are the VM-boot and user-log on phases, as demonstrated in the test results that follow.

The chart that follows shows the contrast between the IOPS demand before and after the two most intensive IO phases as well the performance without IntelliCache. Specifically, this illustration shows how IntelliCache can potentially reduce peak IOPS on shared storage from 1378 to 2.1



*This chart shows how enabling IntelliCache reduces peak IOPS on shared NFS storage from 1378 to, at first, 103 IOPS, and then ultimately 2.1 IOPS.*

Since the two most IO-intensive phases are the boot and logon, we tested the following IntelliCache scenarios:

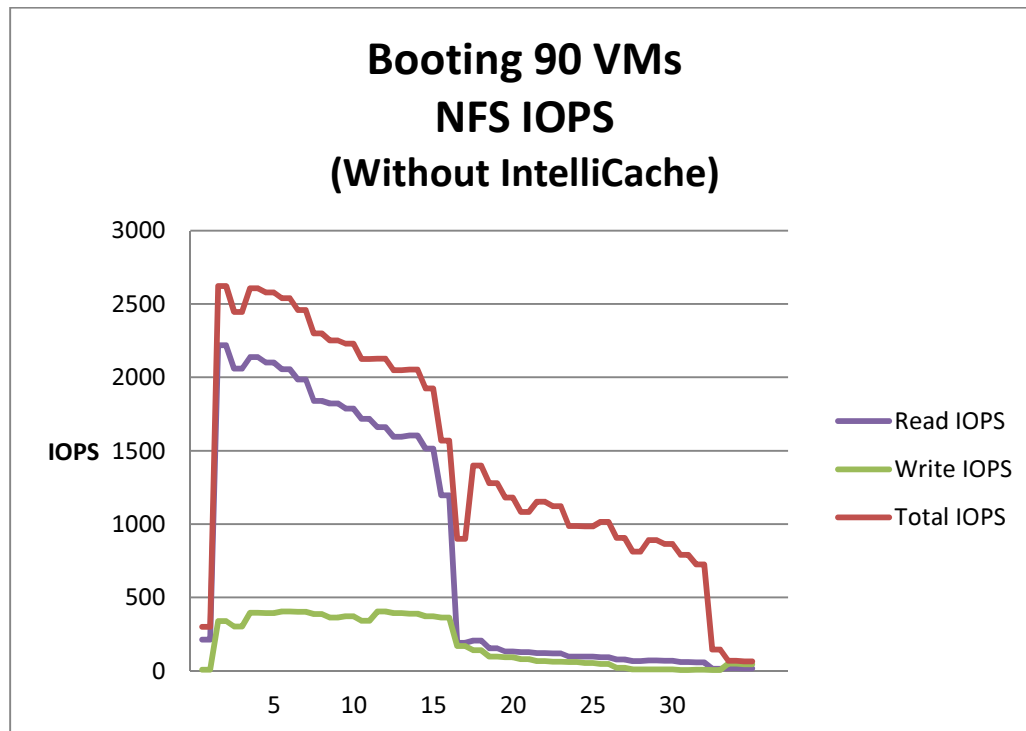
| Test Phase  | VM Lifecycle   |
|---|--|
| Without IntelliCache enabled for the boot process.  | First time the VMs are booted (without IntelliCache enabled).                        |
| With IntelliCache enabled, but before the Read Cache was fully populated in the boot process (the Boot/Cold Cache stage). | First time the VMs are booted (with IntelliCache enabled). (First boot of 90 VMs.)   |
| With IntelliCache enabled and with the Read Cache populated with boot data (the Boot/Warm Cache stage).                   | Second time the VMs are booted (with IntelliCache enabled). (Second boot of 90 VMs.) |
| Without IntelliCache enabled to establish a baseline.   | When the Login VSI test run is run without IntelliCache enabled.                     |
| With IntelliCache enabled, as the Read Cache is being populated during the first Login VSI test run.                      | The first test run of Login VSI with IntelliCache enabled. (Cold Cache.)             |
| With IntelliCache enabled and with the Read Cache populated (the Warm Cache stage).                                       | The second test run of Login VSI with IntelliCache enabled. (Warm Cache.)            |

As expected, we measured dramatically more IOPS on shared storage without IntelliCache enabled and some differences in IOPS between the Cold and Warm Cache phases once IntelliCache was enabled.

For all of the test results that follow, we simulated the user activity (boots, logons) using Login VSI. During the Login VSI test run, users were launched every 30 seconds. In all cases, we measured IOPS on the NetApp storage using NetApp Operations Manager.

## 5.1. Baseline Boot Performance (Without IntelliCache Enabled)

Our test results revealed that without IntelliCache enabled, the shared storage processed significant amounts of IO during the boot phase. At its peak, the shared storage IOPS were recorded at over 2600 Total IOPS during the boot and stabilization phase.



*This graph shows the Read, Write and Total IOPS on the NFS storage as the NetApp Operations Manager recorded during the initial boot test we performed without IntelliCache enabled.<sup>4</sup>*

## 5.2. IntelliCache Enabled: Cold and Warm Cache, Boot

As previously noted, the boot phase of the VM lifecycle is typically the most IO-intensive phase. Like the logon phase, the boot phase has a cold cache and warm cache stage when the Read Cache is populated. Our test results show a vast reduction in IOPS after the Read Cache is populated.

In the first boot test run, we started the VMs and observed the IOPS spike as the Read Cache filled. After the VMs booted, registered, and stabilized with XenDesktop, we shut down the VMs and repeated the test to observe the effect of the warm cache on boot IOPS.

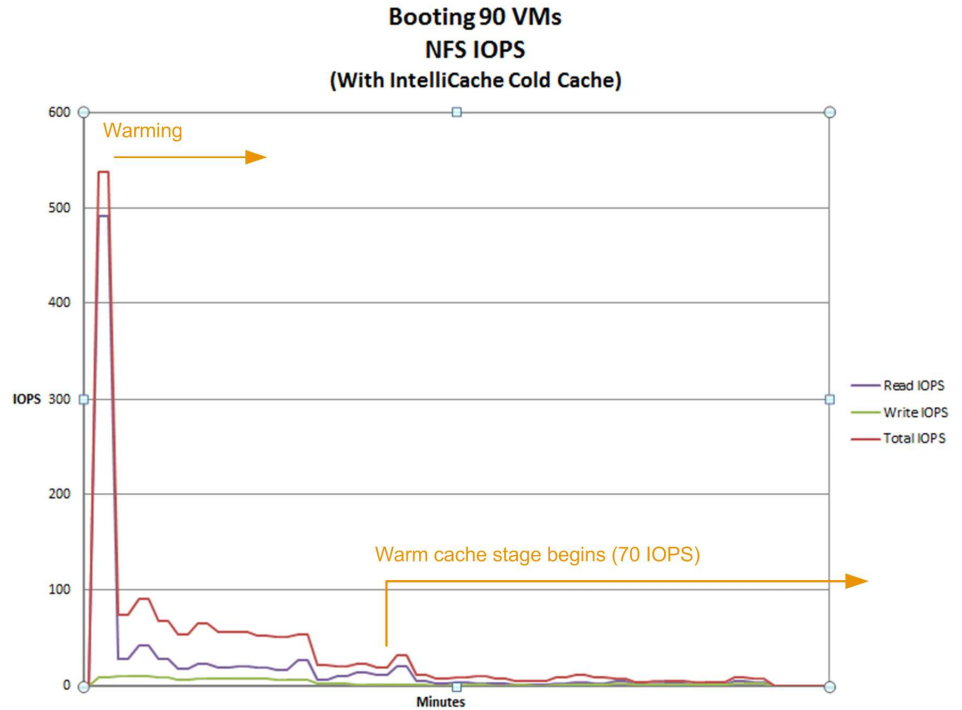
During the first test run, we booted 90 desktop VMs on a XenServer host. After a few desktop VMs connected to the master image and the host began populating its local Read Cache, the cache entered its “warm stage” and IOPS began to fall dramatically (from over 500 IOPS to 70 IOPS in only two minutes). As you can see in the graph that follows, the IOPS in the second graph peak at an initial 70 IOPS and continue to decrease.

<sup>4</sup> The Total IOPS line in this graph represents not only Read and Write IOPS but also Other OPS as reported by NetApp Operations Manager.

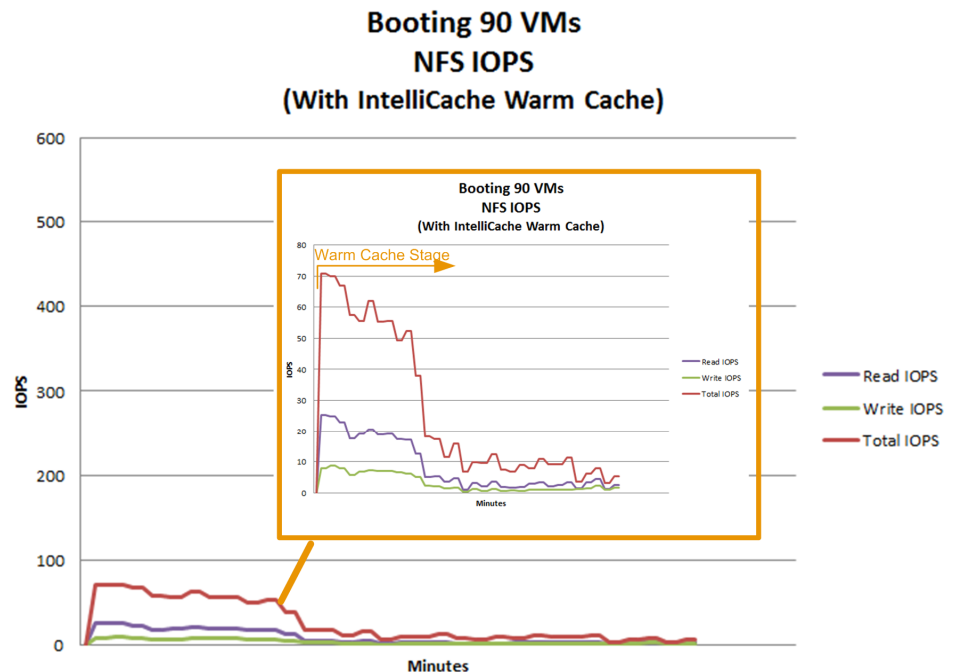
## 1<sup>st</sup> Boot of 90 VMs

This illustration shows how, in the top graph, the initial spike in IOPS as the 90 VMs boot for the first time. As the Read Cache fills, the IOPS decrease significantly and continue to decrease as more data is cached. In the bottom chart, the 90 VMs were booted for the second time and there were significantly fewer IOPS because the cache was populated from the first boot cycle.\*

\*The Total IOPS line in the graphs represents not only Read and Write IOPS but also Other OPS as reported by NetApp Operations Manager.



## 2<sup>nd</sup> Boot of 90 VMs



However, as additional VMs began to boot, some requested different operating-system data and, as a result, had to read the data from the master image on the shared storage. This is shown in the previous illustration by the temporary spike from approximately 70 IOPS back to nearly 100 IOPS in the first graph. After the test reaches the warm cache stage, the desktop VMs sat idle waiting for users to log on and then the VMs were shut down.

The second graph reveals how when the boot test run was executed with a warm cache (for example, when rebooting between shifts of workers), the Read Cache is already warm and populated. As a result, the boot phase begins with approximately 70 IOPS and falls significantly from there.

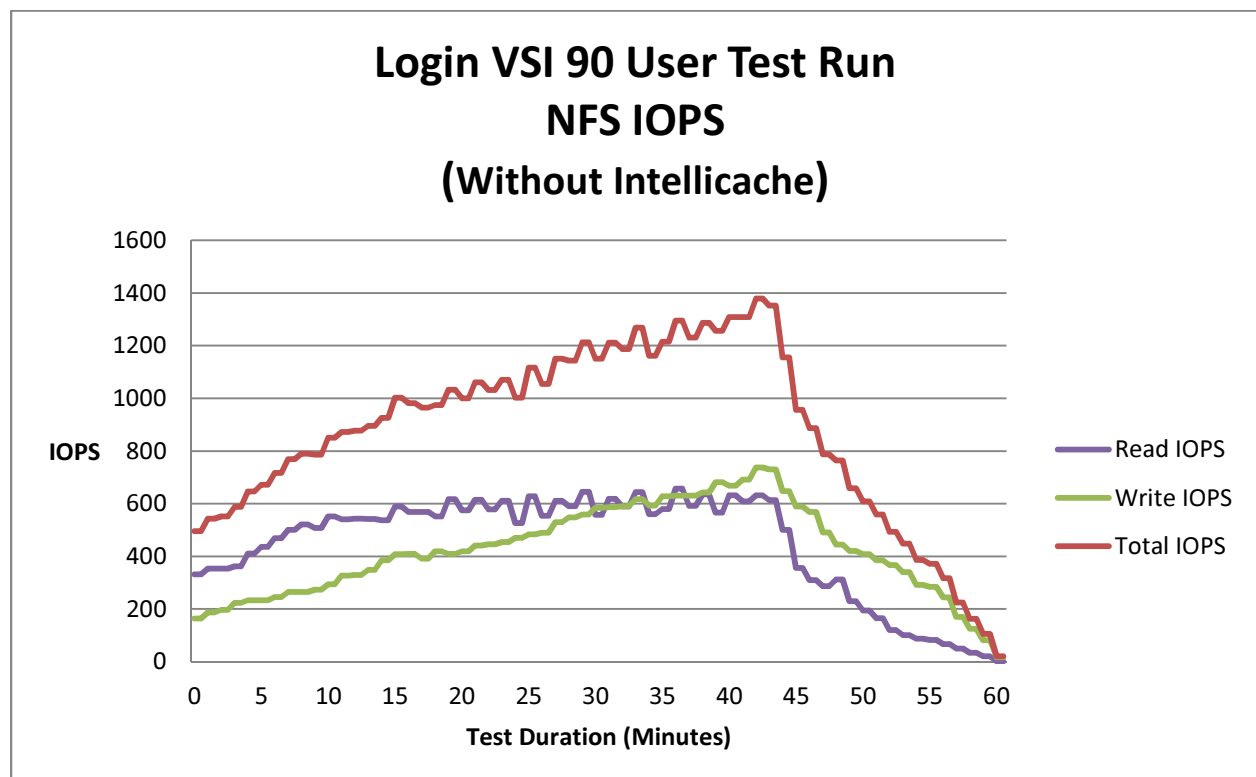
### 5.3. Baseline Login VSI Test Run Performance (Without IntelliCache)

To evaluate the IOPS impact of implementing XenServer IntelliCache, we first had to establish a baseline.

For the initial test scenario, before enabling IntelliCache, we used NetApp NFS shared storage for hosting our 90 Windows 7 desktops. This provided a baseline measurement for the amount of IOPS that the NetApp filer would be required to handle without the use of IntelliCache.

To achieve this, we ran tests with Login VSI 3.0 using a medium workload without IntelliCache enabled. This allowed us to gather IOPS data from the NetApp over the course of the test run. For the 90 user baseline test, we observed Total IOPS (Read + Write IOPS) reaching nearly 1400 IOPS.

The following graph shows the IOPS load on the NetApp filer before IntelliCache is enabled. Note that in the graph the peak IOPS is nearly 1400 IOPS.



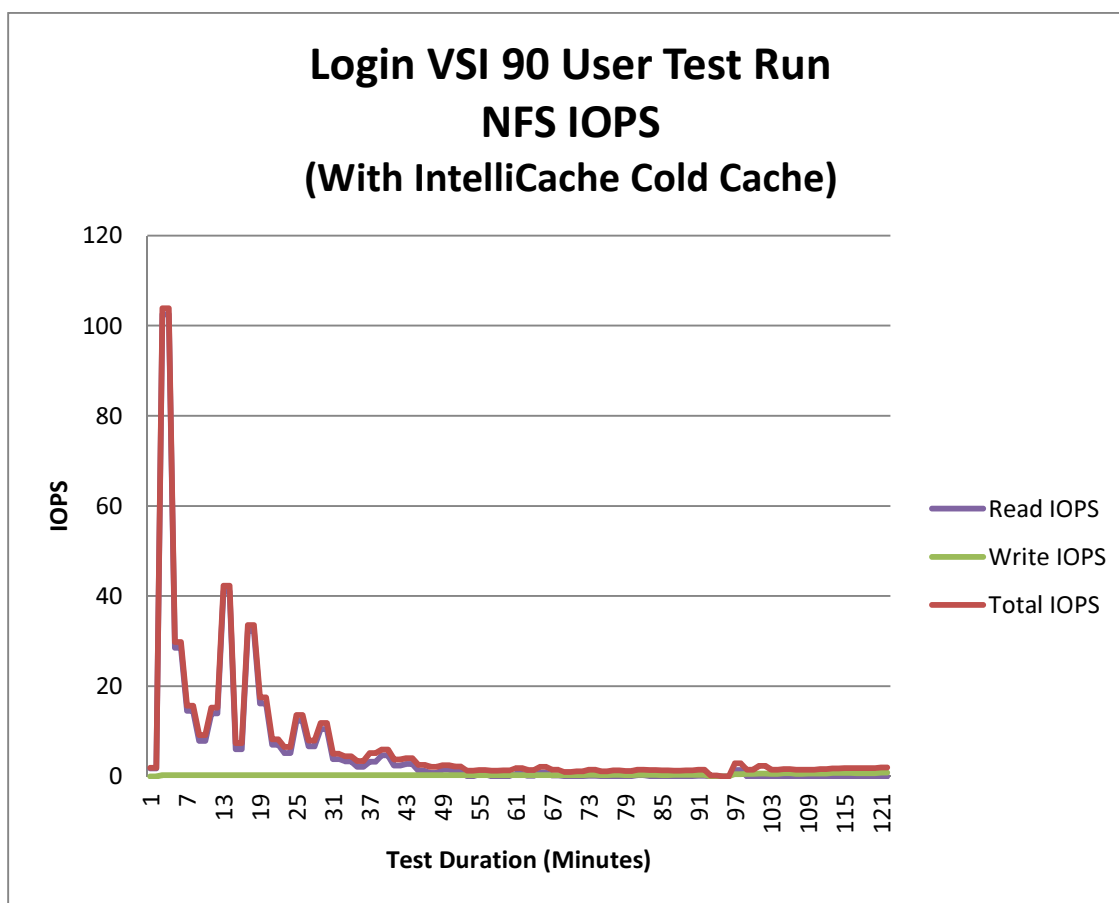
*This graph shows, how without IntelliCache enabled, there are nearly 1400 IOPS on the NetApp storage.*

## 5.4. IntelliCache Enabled: Cold Cache, Login VSI Test Run

Introducing IntelliCache made a significant difference to our test results even during the Cold Cache User Log On phase.

After establishing the baseline, we created a new catalog and desktop group with IntelliCache enabled. Initially, we ran a Login VSI test run with an unpopulated cold cache. Local storage consisted of 2x SSD drives in a RAID 0 configuration. During the Login VSI test run, users were launched every 30 seconds.

As shown in the graph below, as the Read Cache was filling in the user-log on phase, the IOPS on shared storage peaked at 103 IOPS. Typically, the Cold Cache would be populated with operating-system data specific to the log-on process during the period of the initial spike (for example, the 103 IOPS in the graph that follows). This spike only lasts while the Read Cache is being populated (warming). After the first user log on is complete, the cache has most of the data subsequent VMs need. As a result, this peak was only for a few minutes, and then the load dropped below 40 IOPS and continued to decrease over the course of the test run. As shown in the graph below, writes have little to no impact because, in a Pooled Desktop configuration, the desktop VMs write all data to the Write Cache on the local hard drive.



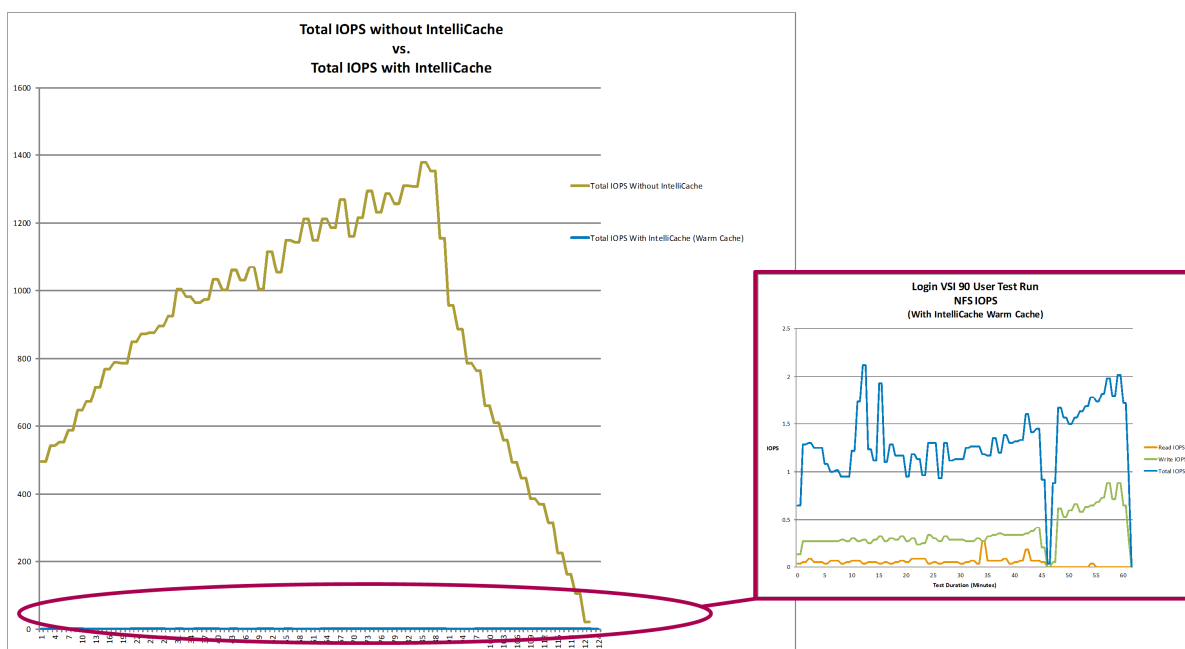
*This graph shows that as the Read Cache fills up, after the first user connects to a desktop, the need to access data from shared storage falls (the load drops below 40 IOPS) and then continues to diminish. All writes are flat because with Pooled Desktops the desktop VMs write their data locally in the Write Cache and not on shared storage.*



## 5.5. IntelliCache Enabled: Warm Cache, Login VSI Test Run

After logging a cold cache test run, the VMs were shut down and then powered back on so we could perform another test run. However, because we had run one initial test, the Read Cache was already populated, which reduced the VMs' need to access data on shared storage. (The second time the test is run the Read Cache persists; however, the VM's Write Caches do not persist.)

The following graph shows how the Total IOPS peak at 2.2 IOPS after the Read Cache is populated. It is important to note that the scale of this graph is only 2.5 IOPS (compared to the 1378 IOPS in the graph based on the scenario without IntelliCache).



*This graph shows that unlike the cold cache scenario, there is no initial spike of IOPS on shared storage. The cache is already populated and therefore, the IOPS reach a peak of 2.2. This is a 98% decrease in peak IOPS compared to the cold cache scenario.*

## 5.6. What about Dedicated Desktops?

This paper focuses primarily on IOPS reductions that you can achieve by configuring IntelliCache for Pooled Desktops since that configuration provides the greatest storage cost savings.

Dedicated Desktops decrease only the read IOPS on shared storage since the VMs still write their persistent data to shared storage. From our testing, when IntelliCache is enabled, we estimate the IOPS savings on shared storage for dedicated desktops to be 30-40%.

Consequently, this paper tested Pooled Desktops since it leads to the greatest reduction in shared storage read and write IOPS.

## 6. Best-Practice Recommendations

Based on the test results we obtained, we make the following best-practice recommendations:

- In cases where you want to reduce IOPS on shared storage, enable IntelliCache for XenDesktop deployments that are virtualized on XenServer.
- Be mindful of how much space you need. In our environment, desktop VMs used 3.2GB for the Read Cache and approximately 700MB for each VM's Write Cache.
  - To determine storage requirements, test a pilot deployment in your environment to calculate reduced storage IOPS requirements.
  - Size your shared storage based on cold cache not warm cache.
- Use XenDesktop hypervisor throttling and power-management features to achieve highest cost savings.
- Be mindful of your local IOPS requirements. Consider using SSDs or ensuring you have an adequate number of SAS drives.
- Use RAID controller with Battery Backed Write Cache when using SAS drives for local storage.

Specific aspects of certain recommendations are discussed in more detail in the sections that follow.

### 6.1. Sizing the Caches to Prevent Falling Back to Shared Storage

When enabling IntelliCache, consider the amount of local disk space required for the Read Cache and the individual VMs' Write Cache files.

Should the local storage reach capacity, IntelliCache will transparently “fall back” to shared storage without end users experiencing a service interruption. To size the local storage needed, Citrix recommends testing in your own environment. Forecasting your local disk space requirements helps prevent XenServer from having to fall back to shared storage to handle the IOPS demand.

In our testing with 90 Windows 7 VMs, we observed a Read Cache size of 3.2GB. To size your Read Cache, it is important that you perform your own testing. Depending on variables in your environment, like patterns of user activity, you may need to plan more space for your Read Cache size.

In addition, your disk-space requirements could increase any time multiple catalogs are present, such as during an upgrade rollout. For example, if virtual machines use multiple versions of the same catalog, Read Cache space usage will increase proportionately.

From a planning perspective, you should assume all of the master image could potentially be stored in the Read Cache. Consequently, if you have multiple catalogs on a host, you should assume that each catalog's master image could be stored in the Read Cache. For example, if you have two catalogs each with different versions of applications in them, both master images could potentially be stored. Likewise, if you are rolling out an operating system update, you may have two catalogs before users reboot and switch over to the new image.

In our Login VSI test runs, each VM's Write Cache was approximately 700MB and all users perform the same actions. However, in a production environment, users might be performing different activities at different times.

## 6.2. Sizing Local Storage to Support IO Requirements

Since IntelliCache relies on storing data on the XenServer host, using SSDs for the Read and Write Caches on each host is a best-practice recommendation. However, your environment may still benefit from IntelliCache provided you have enough local drives to handle the IOPS. These local drives can be SSDs, SAS or, in the case of blade servers, Direct Attached Storage (DAS).

For optimal XenDesktop performance, it is important that the XenServer local storage can handle the IO the virtual desktops generate on the host. If the desktop VMs generate too much IO, VM performance degrades. Consequently, using SSDs may be particularly helpful in environments with blade servers because most blade vendors only provide two slots for local storage per blade. However, DAS drives may also help address this limitation.

During our testing in a lab environment, we used consumer-grade SSDs. However, for performance and reliability in production environments, we recommend enterprise-grade SSDs.

As part of our IntelliCache testing, we also tested IntelliCache with SAS drives. Depending on your host configuration, it may be possible to use SAS drives, provided you use enough of them to handle the required IOPS. Our test results revealed that six 15K SAS drives could support 90 desktop VMs provided the hosts also had a Battery Backed Write Cache RAID controller card. However, it is imperative you size SAS drives correctly. If the SAS drives are unable to handle your XenDesktop workload's IOPS requirements and become an IO bottleneck, performance will degrade and your users will be affected.

The prices for enterprise-grade SSDs have dropped significantly and may continue to fall. While SSDs are more costly than SAS drives, using SSDs still present substantial cost savings over the increased IOPS requirement for shared storage without IntelliCache enabled.

## 6.3. When Using Battery Backed Write Cache is a Best Practice

During our testing, we used a RAID controller with to buffer IO and increase response time to the local disk. Buffering IO data can greatly improve read and write disk throughput for SAS drives; however, it may not be necessary for SSDs.

We tested both SAS drives and SSDs with and without Battery Backed Write Cache:

- SSDs can handle a large number of IOPS and do not need Battery Backed Write Cache. In our testing, 2x SSDs were able to keep up with the IOPS demand that were placed on them.
- SAS drives, however, require using Battery Backed Write Cache due to the relatively low performance of SAS drives for small burst write requests.

When we configured the Battery Backed Write Cache, the RAID controller was left at its default memory configuration since the controller did not support specifying a read-to-write ratio for memory. If your

controller supports adjusting the memory assignment, how much memory you allocate to the controller vs. reads and writes would depend on what aspect of your environment you wanted to improve. For example, if you wanted to improve boot times, you may want to allocate more memory to reads.

## 6.4. Considerations for Existing Deployments

While it is possible to enable IntelliCache in an existing deployment, the ideal time to enable it is when you are first rolling out XenDesktop. The main reason for this consideration is because IntelliCache is enabled during XenServer Setup.

Enabling IntelliCache after Setup may be possible, as described in the *XenServer 6.0 Installation Guide*. However, if you choose to enable it, you must take precautions. By default, XenServer formats its SRs using the LVM format. However, IntelliCache requires SRs formatted with thin provisioning (EXT3). Consequently, if you want to enable IntelliCache and your SR is formatted as LVM, you must destroy and recreate your SR, which results in the data in the SR being erased.

Instead of configuring the entire environment to use IntelliCache, you could alternatively configure any new desktops to use it. In this case, you would create a new XenDesktop Catalog with IntelliCache enabled for the new VMs.

## 6.5. Other Considerations

One limitation that results from using pooled desktops and IntelliCache is that it is not possible to perform live migration (XenMotion) for your VMs. This means VMs must be powered down for routine maintenance.) However, the inconvenience that this causes during maintenance may be outweighed by the cost savings.

## 7. Conclusion

Configuring IntelliCache for XenDesktop deployments virtualized on XenServer can result in significant savings in storage costs. This is a direct result of the sizable reduction in IOPS up to 92%, in both cold- and warm-cache scenarios, when IntelliCache is enabled.

When testing IntelliCache in your own environment, it is important to note three points:

1. To see the full IOPS reduction, you must wait until the Read Cache is largely populated during the user-log on phase. You can tell the Read Cache is largely populated with the data most VMs need when the IOPS level off. In our testing, this took approximately 25 minutes.
2. Shared storage requirements should be sized based on Cold Cache requirements to prevent performance degradation whenever the master image is changed (for example, by applying a Windows Update).
3. When sizing shared storage, be mindful of the number of IOPS that are required when booting during the Cold Cache stage. IOPS requirements can be mitigated by using the XenDesktop hypervisor-throttling and power-management features.

For Pooled Desktop deployments, Citrix recommends enabling IntelliCache provided the environment does not require XenMotion. Enabling IntelliCache provides tremendous value for virtual-desktop environments by lowering the overall Total Cost of Ownership.

## 8. Appendix A: Testing and Configuration

This section provides detailed specifications for how we configured the test environment. It also includes the criteria we used to determine if our tests were successful and information about how we gathered metrics.

### 8.1. Success Criteria for Test Scenarios

For us to consider a test to be successful, it had to meet two different criteria:

1. Login VSI Max Allowed Range
  - The response times Login VSI measured must be within the allowed range Login VSI defined
    - In all of our tests, VSI Max was not reached, as the response times were well within Login VSI's acceptable limits.
2. User Logon Times Below 60 Seconds
  - During a test run, we measured the time it takes once a desktop session is launched for the user to log on and start the Login VSI workload.
    - For a test run to be successful, each user must log on and start the workload in less than 60 seconds.
    - Any user that has a log-on time over 60 seconds was identified as experiencing a performance degradation.

### 8.2. How Metrics Were Gathered

- XenDesktop logon times were gathered using an internally developed tool (known as "STAT") used to launch sessions and record logon time metrics.
- NetApp data was gathered using NetApp Operations Manager.

### 8.3. Test Configuration

Our test results are based on NFS shared storage on a NetApp FAS 3270. Our environment was running XenServer 6.0.2 and XenDesktop 5.6.

We increased the RAM allocated to the XenServer Control Domain from its default of 752 MB to 2048 MB. Increasing memory allocated to the Control Domain is a XenServer best practice for XenDesktop deployments. As described in CTX131047—[XenServer 6.0 Configuration Limits](#), it is possible to increase the Control Domain memory allocation to 2940 MB to support 50-130 VMs. For information about increasing Control Domain memory, see the *XenServer 6.0 Administrator's Guide*.

## 9. Appendix B: Test Infrastructure

This appendix provides information about the test infrastructure, including sections about the configuration of physical and virtual systems.

### 9.1. Physical System Configuration

| Function         | Hypervisor Host  |
|------------------|--|
| Hardware Model   | IBM X3650M3 Rack Server  |
| CPU              | Dual Socket Hex Core CPUs @ 2.93GHz Intel(R) X5670 Xeon(R)           |
| Memory           | 144GB  |
| Storage          | 2x 200GB OCZ Vertex2 SSDs, 350GB RAID 0 Volume                       |
| RAID Controller  | IBM ServeRAID M5015 with 512MB and Battery Backed Write Cache        |
| Network          | 4x Intel 82576 - 2x Bonded for VM Traffic, 2x Bonded for NFS Traffic |
| Operating System | Citrix XenServer 6.02  |
| Misc.            | Increased Dom0 Memory to 2GB   |

| Storage       |                             |
|---------------|-----------------------------|
| System        | NetApp FAS3270              |
| ONTAP Version | 8.0.2P3                     |
| Protocol      | NFS                         |
| Disks         | 17x 15k Spindles RAID-DP    |
| NIC           | 2x 1GB NICs - Multimode VIF |

| Function   | Infrastructure Server 1        |
|------------|--------------------------------|
| System     | Intel 4 Socket Server          |
| CPU        | 4x Intel X7460 6-Core @2.66GHz |
| Memory     | 32GB                           |
| Disk       | 4x 73GB 15K SAS                |
| RAID Level | RAID 5                         |
| NIC        | 6x Intel 82575EB               |
| OS         | XenServer 6.02                 |



|                   |                                |
|-------------------|--------------------------------|
| <b>Function</b>   | Infrastructure Server 2        |
| <b>System</b>     | Intel 4 Socket Server          |
| <b>CPU</b>        | 4x Intel X7460 6-Core @2.66GHz |
| <b>Memory</b>     | 32GB                           |
| <b>Disk</b>       | 4x 73GB 15K SAS                |
| <b>RAID Level</b> | RAID 5                         |
| <b>NIC</b>        | 6x Intel 82575EB               |
| <b>OS</b>         | XenServer 6.02                 |

|                   |                                |
|-------------------|--------------------------------|
| <b>Function</b>   | Infrastructure Server 3        |
| <b>System</b>     | Intel 4 Socket Server          |
| <b>CPU</b>        | 4x Intel X7460 6-Core @2.66GHz |
| <b>Memory</b>     | 32GB                           |
| <b>Disk</b>       | 4x 73GB 15K SAS                |
| <b>RAID Level</b> | RAID 5                         |
| <b>NIC</b>        | 6x Intel 82575EB               |
| <b>OS</b>         | XenServer 6.02                 |

## 9.2. Virtualized System Configuration

|                         |   |
|-------------------------|---|
| <b>Function</b>         | Active Directory Domain Controller                    |
| <b>Hardware Model</b>   | Citrix XenServer 6.02 VM on Infrastructure Server 1   |
| <b>CPU</b>              | 4 vCPU @ 2.66GHz                                      |
| <b>Memory</b>           | 4GB   |
| <b>Storage</b>          | 150GB Local Storage                                   |
| <b>Network</b>          | 1Gbps vNIC  |
| <b>Operating System</b> | Microsoft Windows Server 2008 R2 Enterprise, SP1, x64 |

| Function         | XenDesktop License Server                             |
|------------------|---|
| Hardware Model   | Citrix XenServer 6.02 VM on Infrastructure Server 1   |
| CPU              | 4 vCPU @ 2.66GHz                                      |
| Memory           | 4GB   |
| Storage          | 50GB Local Storage                                    |
| Network          | 1Gbps vNIC  |
| Operating System | Microsoft Windows Server 2008 R2 Enterprise, SP1, x64 |
| Software         | Citrix License Server                                 |

| Function         | XenDesktop DDC  |
|------------------|---|
| Hardware Model   | Citrix XenServer 6.02 VM on Infrastructure Server 1   |
| CPU              | 4 vCPU @ 2.66GHz                                      |
| Memory           | 4GB   |
| Storage          | 50GB Local Storage                                    |
| Network          | 1Gbps vNIC  |
| Operating System | Microsoft Windows Server 2008 R2 Enterprise, SP1, x64 |
| Software         | Citrix XenDesktop 5.6                                 |

| Function         | SQL Server for XenDesktop                             |
|------------------|---|
| Hardware Model   | Citrix XenServer 6.02 VM on Infrastructure Server 1   |
| CPU              | 4 vCPU @ 2.66GHz                                      |
| Memory           | 4GB   |
| Storage          | 50GB Local Storage                                    |
| Network          | 1Gbps vNIC  |
| Operating System | Microsoft Windows Server 2008 R2 Enterprise, SP1, x64 |
| Software         | SQL 2008 R2 Enterprise                                |

| Function         | Citrix Virtual Desktop VM (x90)  |
|------------------|--|
| Hardware Model   | Citrix XenServer 6.02 VM on Hypervisor Host  |
| CPU              | 1 vCPU   |
| Memory           | 1.5GB  |
| Storage          | 24GB Local Storage   |
| Network          | 1Gbps vNIC   |
| Operating System | Windows 7 Enterprise SP1 x86   |
| Software         | <ul style="list-style-type: none"> <li>• Microsoft Office 2010</li> <li>• Citrix Virtual Desktop Agent</li> <li>• VSI 3.0</li> </ul> |

| Function         | STAT Launcher   |
|------------------|---|
| Hardware Model   | Citrix XenServer 6.02 VM on Infrastructure Server 2   |
| CPU              | 4 vCPU @ 2.66GHz                                      |
| Memory           | 4GB   |
| Storage          | 50GB Local Storage                                    |
| Network          | 1Gbps vNIC  |
| Operating System | Microsoft Windows Server 2008 R2 Enterprise, SP1, x64 |

| Function         | ICA Workload Clients (x3)                             |
|------------------|---|
| Hardware Model   | Citrix XenServer 6.02 VM on Infrastructure Server 2   |
| CPU              | 4 vCPU @ 2.66GHz                                      |
| Memory           | 4GB   |
| Storage          | 50GB Local Storage                                    |
| Network          | 1Gbps vNIC  |
| Operating System | Microsoft Windows Server 2008 R2 Enterprise, SP1, x64 |

|                         |   |
|-------------------------|---|
| <b>Function</b>         | <b>ICA Workload Clients (x3)</b>                      |
| <b>Hardware Model</b>   | Citrix XenServer 6.02 VM on Infrastructure Server 3   |
| <b>CPU</b>              | 4 vCPU @ 2.66GHz                                      |
| <b>Memory</b>           | 4GB   |
| <b>Storage</b>          | 50GB Local Storage                                    |
| <b>Network</b>          | 1Gbps vNIC  |
| <b>Operating System</b> | Microsoft Windows Server 2008 R2 Enterprise, SP1, x64 |

## 10. Appendix C: Enabling IntelliCache

Enabling IntelliCache requires performing tasks in two places:

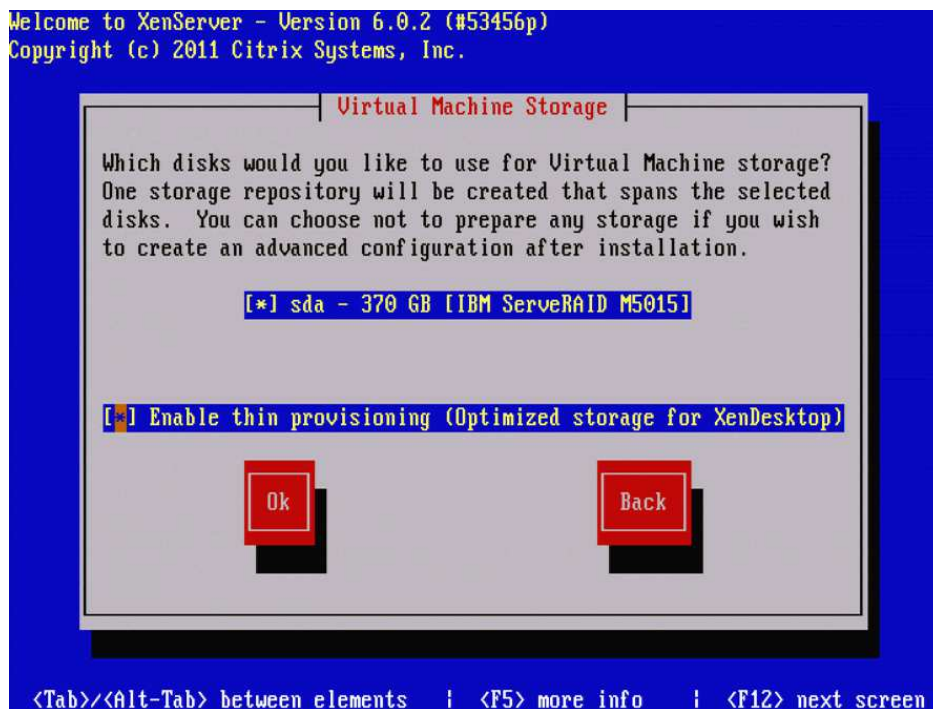
- Thin Provisioning must be enabled on each XenServer host during installation
- IntelliCache must be enabled in XenDesktop when you are adding a host

If, after reading this section, you require more information about configuring IntelliCache, see CTX129052—[How to Use IntelliCache with XenDesktop](#).

**Note:** To use IntelliCache, your shared storage must be NFS.

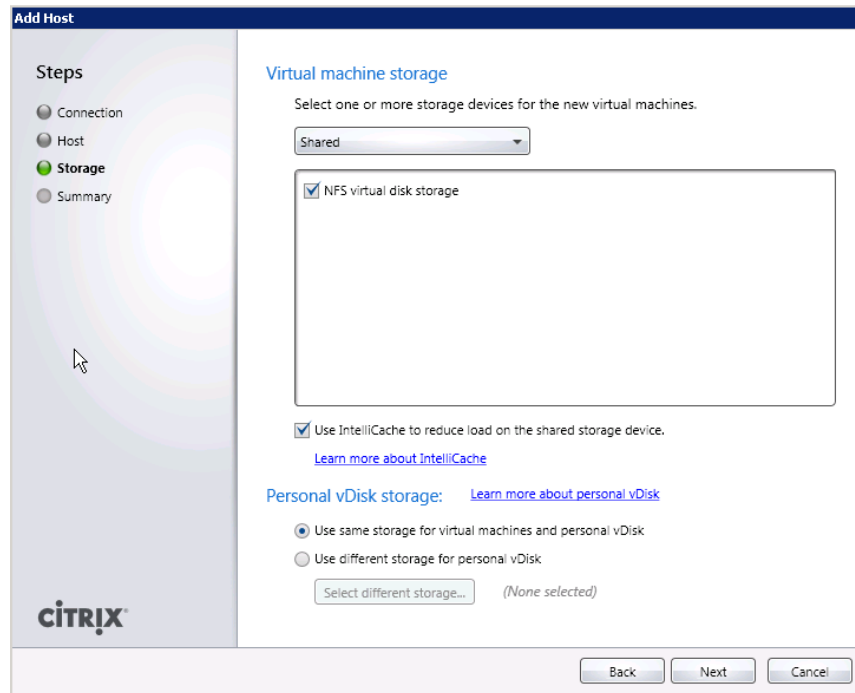
### To enable IntelliCache in XenServer

1. When installing XenServer, select **Enable thin provisioning (Optimized storage for XenDesktop)**.
2. XenServer Setup then creates a Storage Repository which has thin provisioning enabled.



## To enable IntelliCache in XenDesktop

1. When you are adding a XenServer host and you are prompted for the type of storage to use, select **Shared**.
2. Select **Use IntelliCache to reduce load on the shared storage device**.



## 11. Revision History

| Revision | Change Description | Updated By   | Date            |
|----------|--------------------|--|-----------------|
| 1.0      | Document created   | Jeffry Kuhn – XenServer engineering<br>Sarah Vallières — XenServer engineering | August 13, 2012 |



The copyright in this report and all other works of authorship and all developments made, conceived, created, discovered, invented or reduced to practice in the performance of work during this engagement are and shall remain the sole and absolute property of Citrix, subject to a worldwide, non-exclusive license to you for your internal distribution and use as intended hereunder. No license to Citrix products is granted herein. Citrix products must be licensed separately. Citrix warrants that the services have been performed in a professional and workman-like manner using generally accepted industry standards and practices. Your exclusive remedy for breach of this warranty shall be timely re-performance of the work by Citrix such that the warranty is met. THE WARRANTY ABOVE IS EXCLUSIVE AND IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE WITH RESPECT TO THE SERVICES OR PRODUCTS PROVIDED UNDER THIS AGREEMENT, THE PERFORMANCE OF MATERIALS OR PROCESSES DEVELOPED OR PROVIDED UNDER THIS AGREEMENT, OR AS TO THE RESULTS WHICH MAY BE OBTAINED THEREFROM, AND ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR AGAINST INFRINGEMENT. Citrix' liability to you with respect to any services rendered shall be limited to the amount actually paid by you. IN NO EVENT SHALL EITHER PARTY BE LIABLE TO THE OTHER PARTY HEREUNDER FOR ANY INCIDENTAL, CONSEQUENTIAL, INDIRECT OR PUNITIVE DAMAGES (INCLUDING BUT NOT LIMITED TO LOST PROFITS) REGARDLESS OF WHETHER SUCH LIABILITY IS BASED ON BREACH OF CONTRACT, TORT, OR STRICT LIABILITY. Disputes regarding this engagement shall be governed by the internal laws of the State of Florida.