ZFS STORAGE APPLIANCE

How to Migrate Data to Oracle ZFS Storage Appliance Using Shadow Migration

ORACLE WHITE PAPER | OCTOBER 2016

Table of Contents

Introduction

Oracle ZFS Storage Appliance is an application-engineered storage system designed to provide unique Oracle software integration with storage performance and efficiency.

One of the tools available to the Oracle ZFS Storage Appliance administrator is the Shadow Migration feature. Shadow Migration allows the administrator to move data from legacy systems via NFS and to serve the file systems that were previously held on a legacy system, or systems to the same client base, with the added features and abilities unique to Oracle ZFS Storage Appliance.

For the purpose of this document the file systems are defined as follows:

» Legacy file system. This is the definitive data source at the beginning of the data migration process that resides on legacy storage.
» Replacement file system. This is the migration destination that typically resides on the replacement storage.

After the migration process has successfully completed, the replacement file system becomes the definitive data source and the legacy file system storage can be repurposed or taken out of service.

Comparing the Methods for Migrating Data

There are traditionally two methods employed for data migration projects: synchronization or interposition. In the following section, these different methodologies and their main advantages and disadvantages are discussed.

Synchronization Method

Synchronization is typically implemented when copying data directly from the legacy source to the replacement source, while the data is still being accessed by clients through the primary route.

Figure 1 shows a typical synchronization configuration.

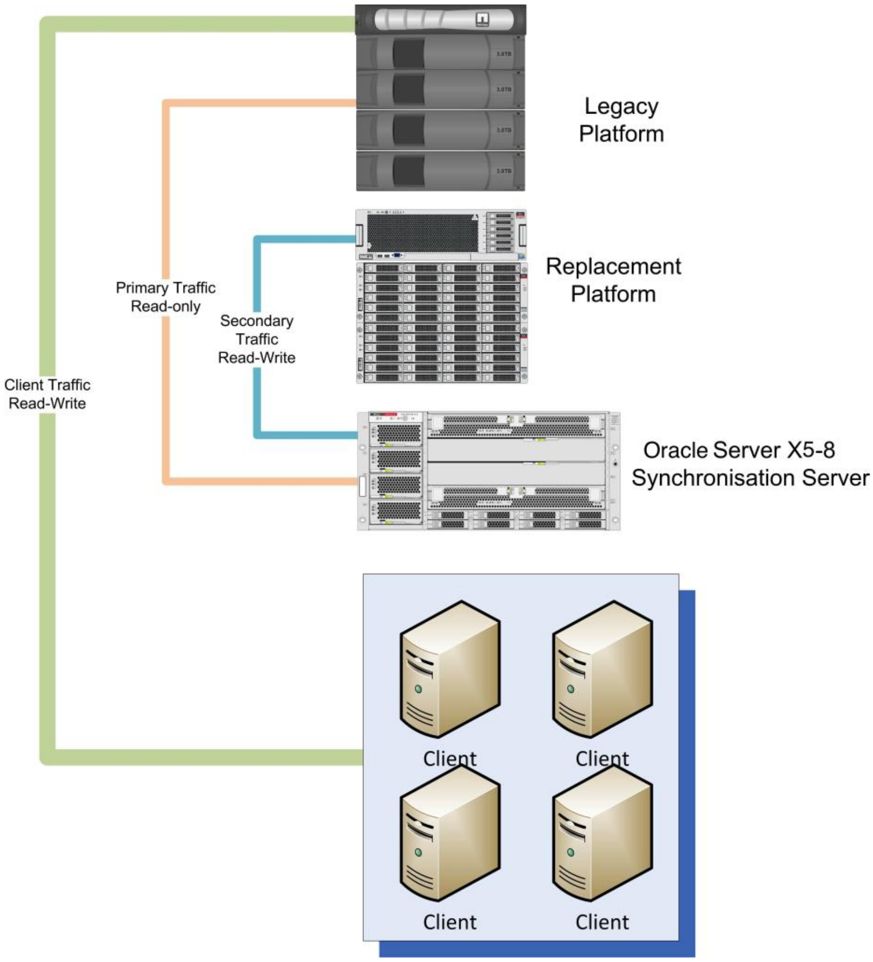

Figure 1. Architecture for data migration using synchronization.

Synchronization utilities typically examine the state of data in both legacy and replacement sources to determine the minimum amount of data to be copied. Only changes that have occurred since the last run of the synchronization tool need to be copied: new and updated files copied across, deleted files removed, new directories created, and deleted directories are removed from the replacement source.

Some synchronization utilities divide the files to be copied into chunks which are then checksummed on both the legacy and replacement sources. These checksums are compared to determine the minimum amount of data to be copied. If the checksums match, the chunk doesn't need to be copied from the legacy source to the replacement source. If the checksums do not match, the chunk is copied across the network and applied to the replacement file.

When the synchronization run is initiated, a "plan of action" is created for the list of files that need to be copied from the legacy to the replacement file systems, as shown in figure 2. The dashed lines show the files that are in the plan to be copied to the (until now) empty replacement file system.

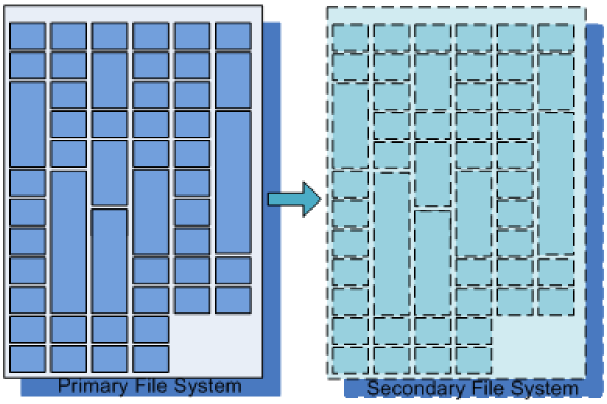

Figure 2. File copy "plan of action."

After the plan has been created, files are copied between the legacy and replacement file systems. Any additions to the legacy file system won't form part of the "plan of action" and therefore will not be copied between the file systems.

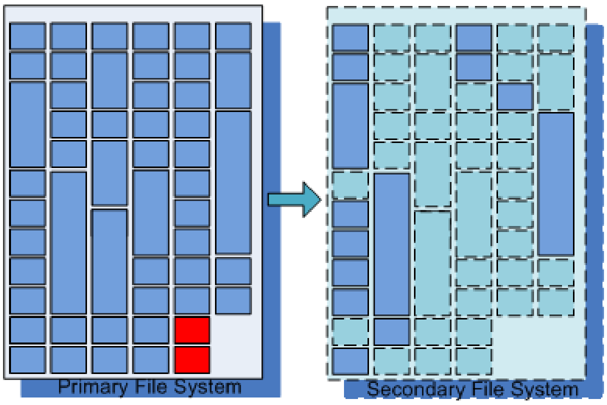

Figure 3. Synchronization "copy" phase.

As shown in figure 3, two new files are created on the legacy file system (depicted as red boxes). There are no corresponding boxes in the "plan of action."

If the legacy data source changes either during or after the file selection process, the changes are effectively lost for the duration of the synchronization run.

Any legacy file system file included in the "plan of action" that is changed will have the changes copied across only if the changes are made to the legacy before the copy phase reaches the file. If any changes are made after the synchronization process has dealt with the file, these changes are lost.

It might be necessary to run the synchronization process multiple times to ensure that these previously missed changes are captured and dealt with.

Thus, the synchronization process is an iterative one with the number of iterations defined by the volatility of the file source.

Eventually, however, it will be necessary to stop access to the legacy source to ensure all changes have been applied to the replacement source. For this to be completely successful, no update actions should be allowed to the legacy source. If updating access times to the files in the legacy source is necessary, for auditing or compliance requirements, disallow all access including read-only access.

During this downtime period, the last synchronization should occur and the synchronization process applied until no changes are required to be made between the legacy and replacement sources.

Once the synchronization run comes up “clean,” disable the legacy source and all of the clients should then refer to the replacement source as the authoritative version.

By swapping IP addresses for the sources, the administrator can accomplish the switch in a way that reduces the burden on the administration staff having to change all the clients to refer to the new authoritative source.

Note: For all of the files to be migrated, they must be readable by the interposer and thus, the interposer device must have effective `root` access to the legacy file system. Also, to ensure that the permissions and access control lists for the files and folders are duplicated on the replacement file system, the legacy and replacement platforms and the interposer must have the same name service configuration.

Advantages of Synchronization Data Migration Method

The primary advantage of the synchronization method is that there are two copies of the data in the legacy and replacement sources after the initial synchronization process. The replacement source might not be as up to date as the legacy as it is only as up to date as the last synchronization run, but this may sufficient for disaster recovery policy.

The synchronization process can be run multiple times to reduce the amount of data required to be copied during the final synchronization downtime. The “go” decision for the switchover needs only to be made just prior to the final synchronization runs.

Synchronization Disadvantages

Except for the initial run, the duration of all the synchronization process runs depends entirely on the volatility of the file system. The downtime required for the final synchronization might be difficult (or even impossible) to quantify in a meaningful way. In the worst case, the entire file system contents might have changed with additional files being created, which means that the final downtime is more than that required for the initial synchronization run.

Remember that during this final copy, client access should not be allowed, which might not be tolerable for the downtime window allowed.

Another disadvantage in using the synchronization method is that until the final downtime synchronization run, clients will continue to use the older storage and/or servers while the replacement storage and/or servers with their additional features and services remain effectively idle. In other words, the return on investment won’t start until after the final synchronization process run.

If the legacy platform is unable to perform the synchronization itself, an added network hop is incurred by the synchronization server having to read the source data over the network and then resend the same data to the replacement platform.

Interposition Method

When using the interposition methodology, a device is placed between the legacy source and the client users of the source. The interposing device then performs the migration from legacy to replacement sources in the background, transparently to the clients.

While the interposer is in place, the legacy file system must not change. Any changes made will either be ignored or worse, cause corruption of the file in question, depending on at what point the change is made to the legacy file system. Ideally the legacy file system must not be shared with any other devices other than the interposer.

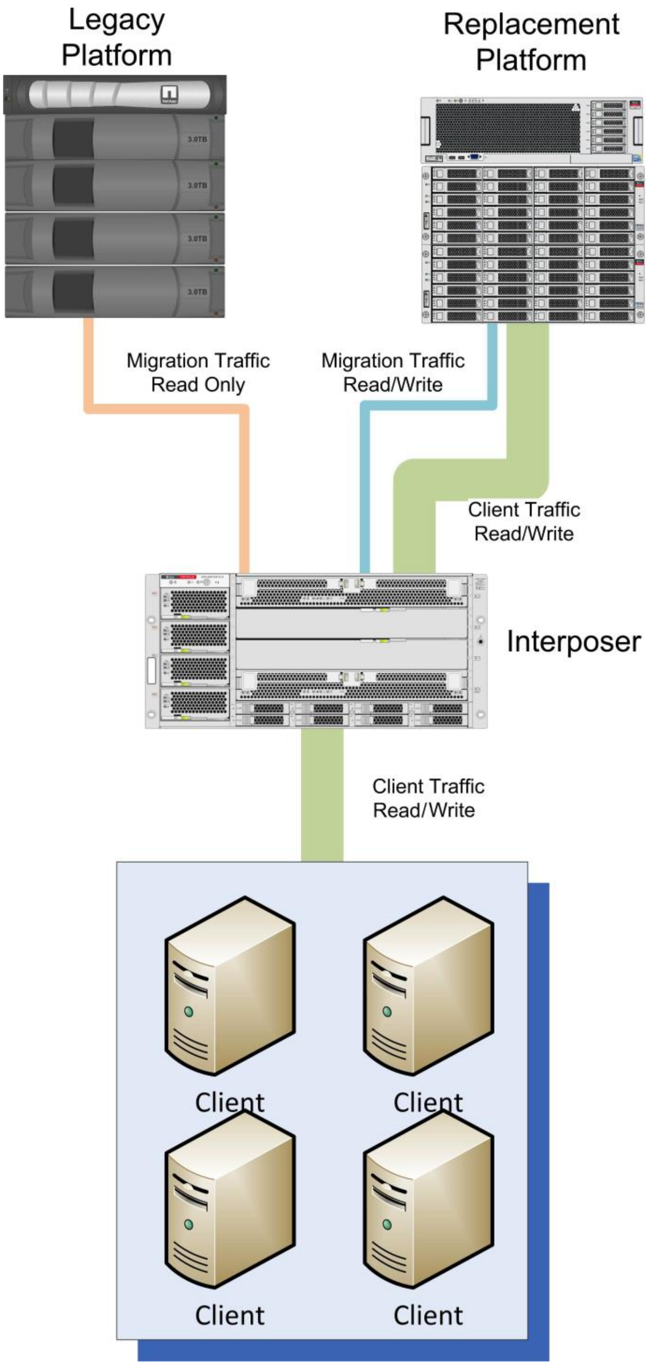

Figure 4. Architecture for interposition method of data migration.

Downtime is required to point the clients to the interposing equipment at the start of the migration process.

During the migration process, the interposing device must select the data source, depending upon whether the data in question has already been migrated from the legacy to replacement source and the type of access requested (read only or read/write).

If the file has been copied across to the new platform, any I/O request is satisfied from there. If the file has not been copied to the new platform, the migration process will first migrate the file to the new platform and then allow the I/O request to progress, served by the new platform.

This methodology, therefore, can potentially add some latency to an I/O request, but only if the file has not already been migrated through the automatic process or has been the subject of an I/O request since the migration has occurred. If the subject is large, this might not be a tolerable approach due to the additional latency incurred by migrating the file prior to allowing the I/O request to progress.

For small files, this might not be an issue because smaller files should copy over relatively quickly. If a given application requires a large number of small files, the first time the application accesses these files could entail a considerable added latency.

Once the migration has completed, the interposing equipment can be removed. Downtime might be required to refer the clients to the fully-migrated replacement source, which is then the authoritative copy of the data.

When the interposition migration is initiated, the clients access the legacy file system “through” the replacement file system. The replacement represents all the files that exist in the legacy file system, as though they existed in the replacement file system, even though no data transfer has actually occurred. This is depicted in figure 5 as dotted file outlines.

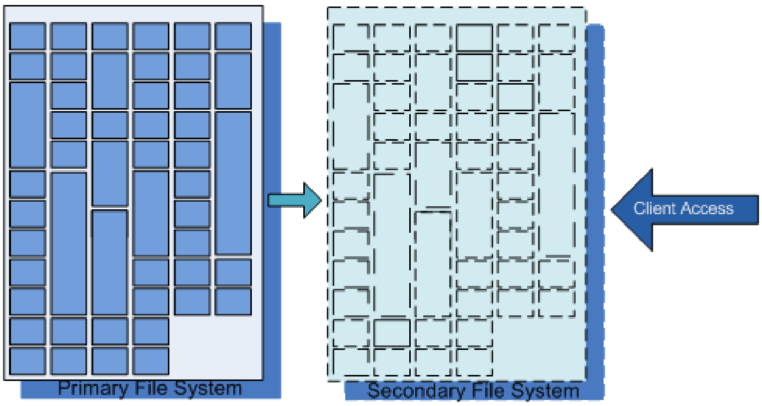

Figure 5. Interposition initiation.

As the migration progresses, files are copied from legacy to replacement file systems. A request to read a file causes one of two sequences to occur, depending on whether the file has already been migrated from the legacy to the replacement file system or not:

» If file has already been migrated to the replacement file system, the I/O request is processed without delay and is satisfied from the replacement file system. This case is shown in figure 6.

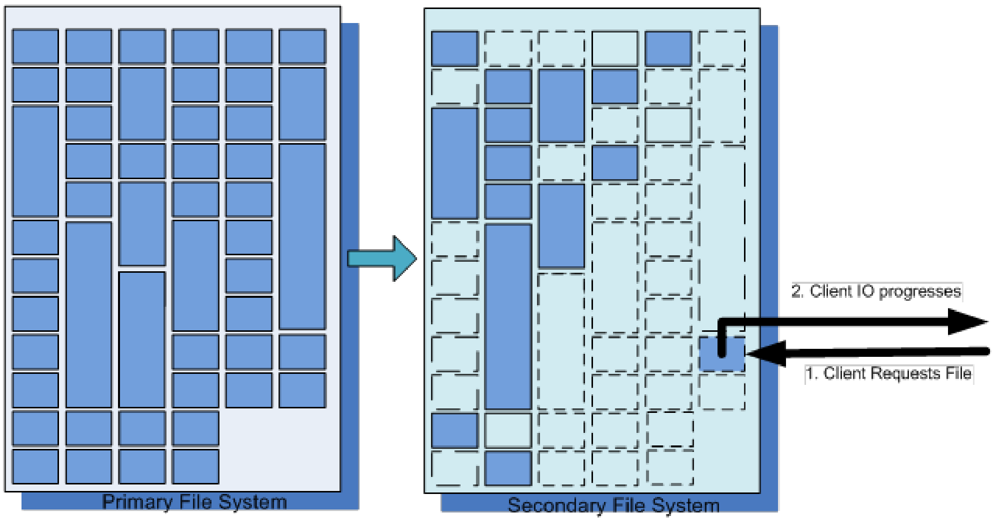

Figure 6. Interposer read—Case 1: The file has already migrated to replacement file system.

» If the file has not yet been migrated, the I/O request is paused while the interposer initiates the file migration. After the file migration has completed, the I/O request is allowed to progress and is satisfied from the replacement file system. This case is shown in figure 7.

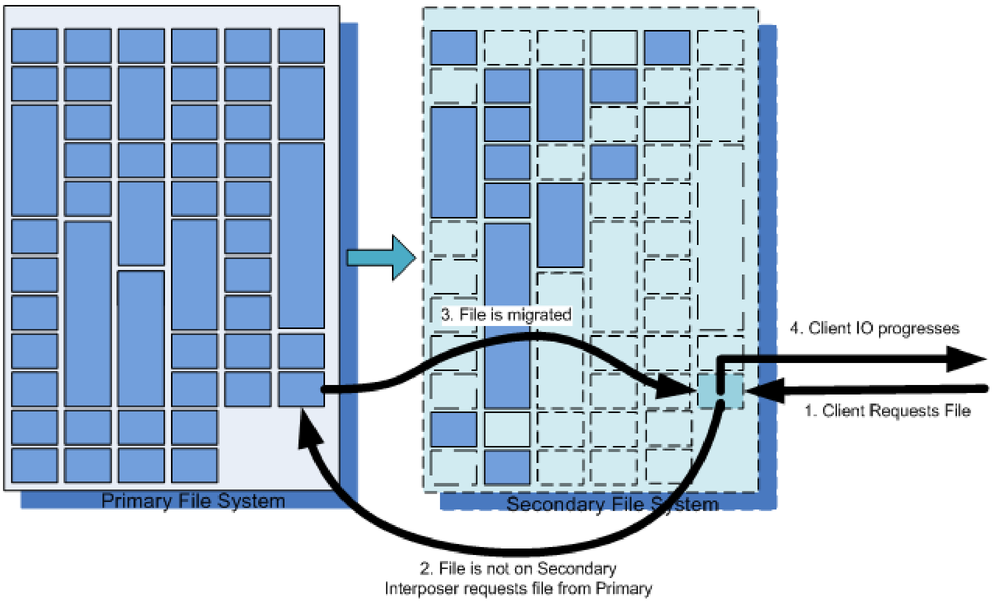

Figure 7. Interposer read—Case 2: The file has not yet migrated from primary/legacy file system.

Any modification of a file causes the migration process to copy the file from the legacy file system to the replacement file system (assuming it has not already been copied). The modification is performed on the replacement file system. Similarly, any new files are created on the replacement file system without any change made to the legacy file system. Figure 8 shows this situation with the new files shown as red boxes.

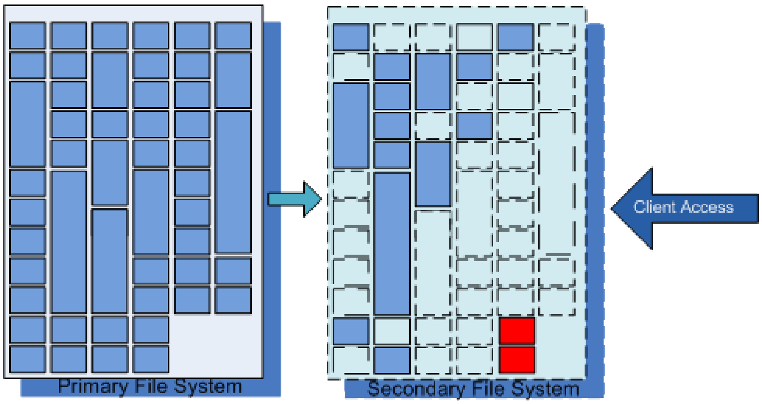

Figure 8. Interposition in progress, showing new files created directly on secondary/replacement file system.

All the while, client access is being served by the replacement file system. After all of the files are copied across, the interposition migration automatically stops and the relationship of legacy and replacement file systems no longer need to be maintained.

Advantages of Interposition Method of Data Migration

After the data migration has started, the features and services of the destination platform are available to the clients.

The only downtime required is the time it takes to modify the client configuration to refer to the interposition device as the client access path and, possibly, the removal of the interposition device and further client configuration to refer to the replacement.

Interposition Disadvantages

Interposition devices tend to be left in place for a long time after the migration projects have finished due to the required downtime needed to refer clients to the replacement storage platform. The indirection through the interposer adds some latency, albeit very small for each request. Over the lifetime of the project, this can add up to a large amount of time. After the migration has finished, this additional latency is effectively wasted time unless the interposer device provides additional data services that are not available on the replacement platform.

The point at which the commitment to the migration project must be made is exactly when the interposer device is placed in the data path between the clients and the legacy and replacement platforms. Once the device is in place, it is difficult to undo all of the work and to ensure that all of the newly added and modified files that reside only on the replacement platform are copied back to the legacy platform.

Which Method is Best?

The only possible answer to this question is, “It depends!” The decision depends on the access patterns of the file system in question and the distribution of file sizes of that file system. It is possible to deploy both methods in a data migration project—although obviously both methods cannot be used on the same file system.

When very large files exist, the synchronization approach might be more acceptable, particularly where the files are accessed in read-only mode. When a large number of relatively volatile files exist on the migration source, the interposition mode might be more tolerable than the synchronization method due to its quantifiable downtime requirements.

Unfortunately, there is no perfect method that caters to every circumstance. Local policies and access patterns should determine which method has the least impact for each file system being migrated.

Accomplishing Data Migration Using the Oracle ZFS Storage Appliance Shadow Migration Feature

Oracle ZFS Storage Appliance can function as an interposer device with a major advantage: not requiring removal at the end of any migration process. Oracle ZFS Storage Appliance provides this interposition migration feature through its service known as Shadow Migration.

Shadow Migration can be used to migrate data from legacy platforms using NFS and to migrate data between storage pools when the source is defined as local to Oracle ZFS Storage Appliance.

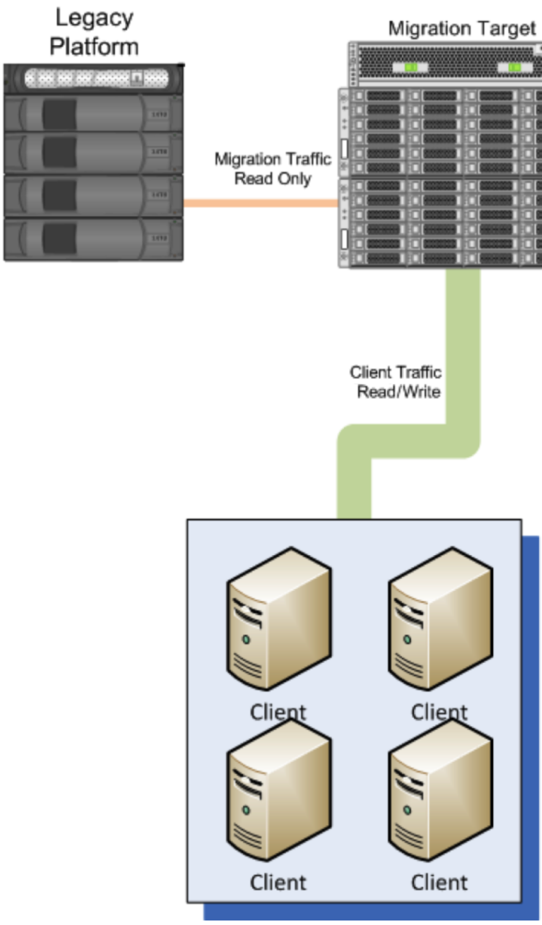

Figure 9. Oracle ZFS Storage Appliance Shadow Migration feature.

Using Shadow Migration to Move Data from an NFS Source

When using Oracle ZFS Storage Appliance to migrate data from a legacy source, ensure that the performance provided by the legacy platform is configured to provide as much bandwidth as possible to Oracle ZFS Storage Appliance. This means ensuring that all client access is redirected to Oracle ZFS Storage Appliance and that any network links, between the legacy platform and Oracle ZFS Storage Appliance, are contention free and configured to offer the maximum bandwidth possible. In practical terms, this can be achieved by placing a dedicated Ethernet switch between the platforms and servicing only them.

Where possible, allocate multiple interfaces on each platform for the duration of the migration process. But any configuration must allow client access to Oracle ZFS Storage Appliance which, having replaced the legacy platform, is now the client-side service. Also, configure any file systems being migrated as read-only as stated earlier.

Figure 10 illustrates a possible physical network configuration during the migration.

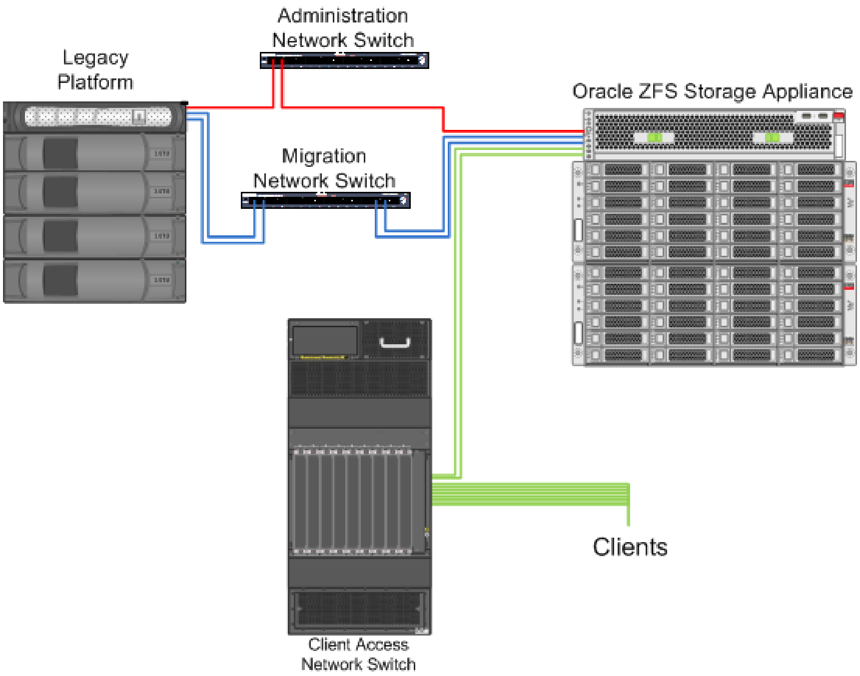

Figure 10. Possible physical networking configuration.

To disable client access, remove the legacy platform from any switch or router to which the clients have access. As a result of this move, it might be necessary to reconfigure the IP addressing on both platforms to reflect the temporary network created to support the migration process.

The clients will either need to be reconfigured to use the services of the replacement Oracle ZFS Storage Appliance or, alternatively, Oracle ZFS Storage Appliance can be configured to take the networking addresses of the legacy equipment.

Performance Considerations

The performance of Shadow Migration depends on a number of factors:

» File size profile
» Number of files
» Network contention on the migration network
» Network configuration of the migration network
» Destination storage pool data protection configuration
» Log and cache configuration
» Number of devices in the destination pool
» Performance of the legacy device
» Volatility of file system during migration

» Access pattern of existing data during migration

The first step to providing the optimum migration performance is to ensure that the network path between the source and destination devices is free from contention and that the network interfaces and devices provide the highest possible bandwidth. This might mean temporarily using a network switch to carry only the migration traffic, as shown in the figure 10 configuration.

Continue to maintain administration access on both legacy and replacement platforms to allow administrators to allow administrators to monitor performance and any hardware failures that could occur.

It might also be possible to configure multiple interfaces as a single virtual interface—in the Oracle ZFS Storage Appliance family, this is configured as IP multipathing (IPMP) or by configuring an LACP aggregation link interface. To get the benefit from these configurations, the legacy platform must also support a similar configuration. Available methodologies and their terminology depend on the manufacturer.

Before allocating all of the network devices to the migration process, consider the client access requirements during the migration process.

The file size profile can affect the migration performance markedly—a large number of very small files will take longer to migrate than the same overall size of very large files. This is due to the overhead of creating the files with the correct attributes and also the simple act of opening and closing each individual file.

To give an example of how the migration times can vary, a test was run to measure the time taken to migrate 64 GB of data based on the following file profiles:

TABLE 1: MIGRATION TEST FILE PROFILES

File Profile	Number of Files	Description
GB	128	Identical files all of 1,073,741,824 bytes.
MB	65,536	Identical files all of 1,048,576 bytes.
Mx (for mixture)	1,070,184	Mixture of “real-world” application files with an average size of 1,410,187 bytes and a standard deviation of 19,018 bytes. Range of 0 bytes to 2,146,435,072 bytes.

In addition to varying the profile of files being migrated, each test was run with a different number of threads (64, 32, 16, and 8) which is set in Configuration > Services > Shadow Migration.

Deduplication and compression were turned off and file systems were migrated from the same set of disks and to the same pool configuration to avoid adding any further variables.

The amount of data to be transferred was the same in each of the above profiles, although the sizes of resulting shares varied slightly due to the differences in directory sizes because of differing number of files.

Different NFS versions of the source were also tested to show possible variations. The NFS client mount parameters cannot be modified on Oracle ZFS Storage Appliance and the defaults or negotiated values must be used.

Finally, the maximum transmission unit (MTU) size varied from the default of 1,500 bytes to 9,000 bytes—the maximum available for Oracle ZFS Storage Appliance.

Note: Resetting the MTU is not a trivial decision. It can affect performance adversely in an environment where there is a lot of contention or collisions, or where the cabling is affected by interference that can cause corruption of the data stream due to the requirement to retransmit errant packets. In this example, the deployed switch only carried the migration traffic, and the cabling runs were extremely short.

The switch used to host the migration traffic was set at 10 Gigabit/sec full duplex for all test runs.

The graph shown in figure 11 shows the overall time taken to migrate the 64 GB of data and figure 12 shows the average throughput achieved for each configuration. The point of these figures is to show the relative differences incurred as some of the variables are modified, not the absolute time and bandwidth.

The maximum file size chosen was just under 2 GB due to the restrictions on file sizes supported by NFSv2.

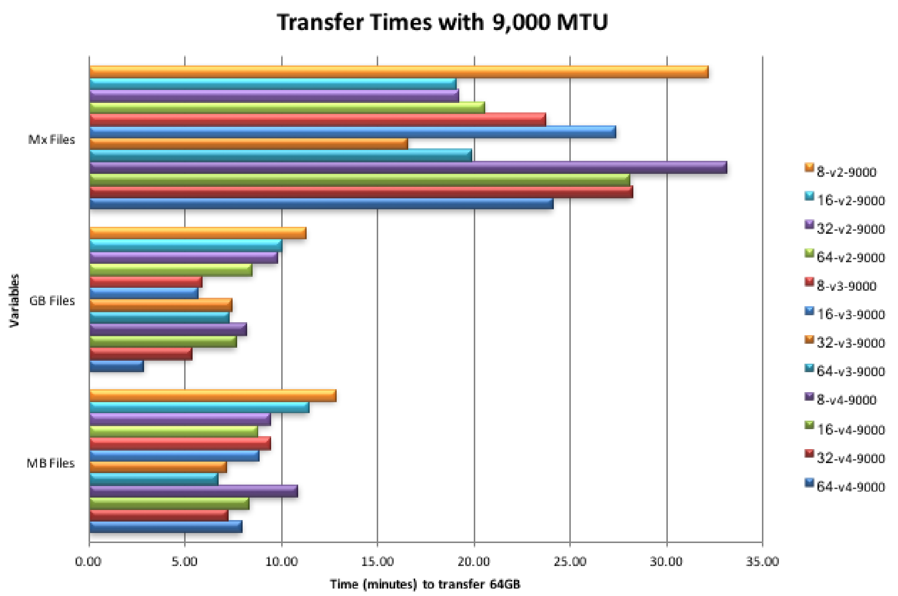

Figure 11. Time taken to migrate 64 GB of data using Shadow Migration with MTU at 9,000.

The naming of the data series is formed from:

<*number of threads*> -v <*nfs version*> - <*MTU*>

Another way of looking at the above data is the throughput achieved by the different configurations. In this case, this is effectively the inverse of the “time” graphs due to the identically sized data sets.

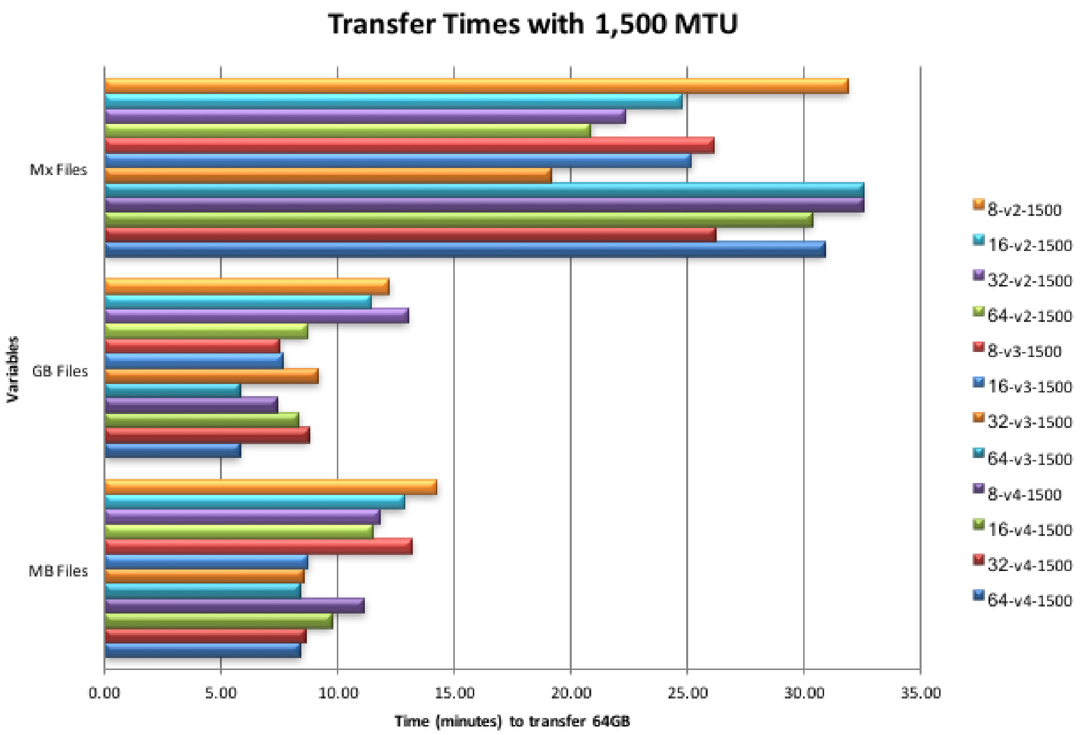

Figure 12. Time taken to migrate 64 GB of data using Shadow Migration with MTU at 1,500.

As the figures show, performance varies depending on the file size distribution and networking configuration. Changing the MTU on the networking interfaces does make a difference in the transfer times, but there is no guarantee that increasing the MTU will increase the throughput, particularly in a high-contention environment. In this type of environment, increasing the MTU requires retransmitting packets of 9,000 bytes rather than 1,500 bytes.

The following figure shows the percentage difference in transfer rates differences between an MTU of 1,500 bytes and 9,000 bytes.

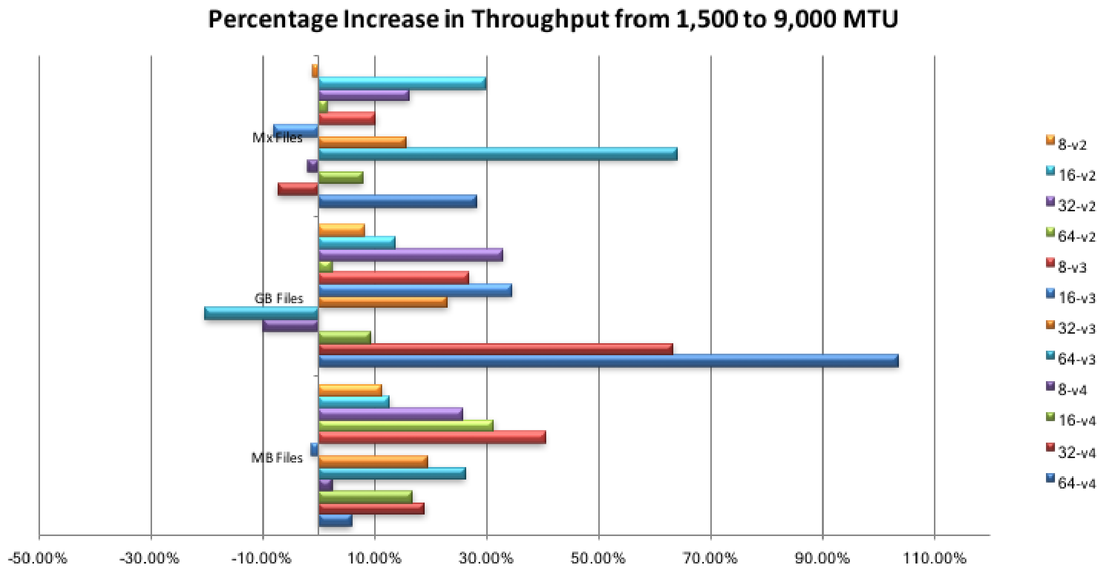

Figure 13. Percentage difference from 1,500 to 9,000 MTU.

Configuring the legacy platform for NFSv4 should be the priority, especially where complex access control lists ACLs) are deployed and need to be maintained. For these ACLs to be meaningful, the legacy storage and Oracle ZFS Storage Appliance must have access to and be configured to use the same directory services: LDAP, Active Directory, Network Information System (NIS), and so forth.

Log and cache devices within Oracle ZFS Storage Appliance can make a significant difference to the performance of migration performance as well as performance of data access by clients during migration. Shadow Migration performs all writes synchronously, which benefits greatly from the application of log devices.

Figure 14 shows a comparison of Shadow Migration runs in which the only difference is the use of log devices.

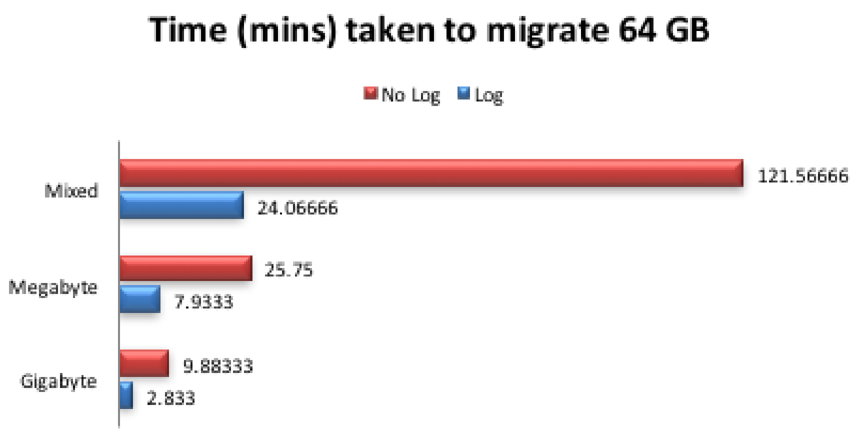

Figure 14. Time taken to migrate 64 MB of data with and without log devices.

Using Shadow Migration to Move Data from a Local Source

Shadow Migration also can be used to migrate data from one storage pool to another within an Oracle ZFS Storage Appliance. This can be a useful tool to move data from a high performance pool to a high capacity one—for example, moving from a RAID 10 pool to a RAID-Z2 pool—when data still needs to be retained, but where access patterns have changed such that the data doesn't require the high-performance support.

Shadow Migration also can be used to change settings that cannot otherwise be modified, such as the file system record size or creating a compressed version of the file system.

Regardless of use case, the method of data migration, however, is the same. The source file system must only be accessed in read-only mode (or not at all) by the clients, as Shadow Migration is not designed to handle dynamic source file systems.

Continuing the performance tests using the same file size distributions used for NFS-based Shadow Migration, local migrations were undertaken without changing any of the volume settings and to the same pool as the source.

Once again, the following results should be viewed as relative times for the varying file size distributions and not as an absolute performance example.

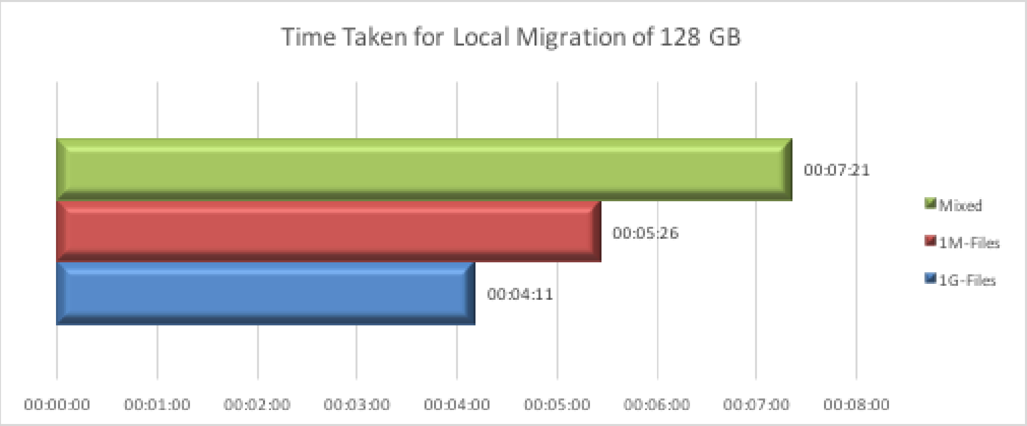

Figure 15. Time taken to migrate 128 GB of data from a local source.

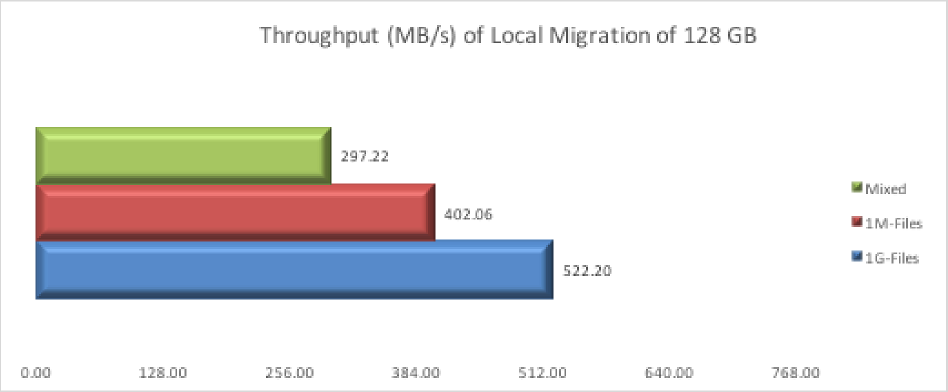

Figure 16. Throughput to migrate 128 GB of data from a local source.

Monitoring Shadow Migration Using Analytics

Oracle ZFS Storage Appliance provides robust analytics which allow the storage administrator to view the current state of the Oracle ZFS Storage Appliance performance and client I/O requests along with network, disk, and CPU load. The detailed browser user interface (BUI) performance metric displays allow drill-down views to individual components within Oracle ZFS Storage Appliance, providing unprecedented access to the end-to-end performance characteristics.

Shadow Migration has a number of available performance metrics:

- Shadow Migration bytes:
  - Broken down by file name
  - Broken down by project
  - Broken down by share
  - As a raw statistic
- Shadow Migration ops:
  - Broken down by file name
  - Broken down by project
  - Broken down by share
  - Broken down by latency
  - As a raw statistic
- Shadow Migration requests:
  - Broken down by file name
  - Broken down by project
  - Broken down by share
  - Broken down by latency
  - As a raw statistic

Aligned with these performance indicators, the network load also can be an important factor.

Figure 17 shows an analytics trace of three concurrently migrated volumes.

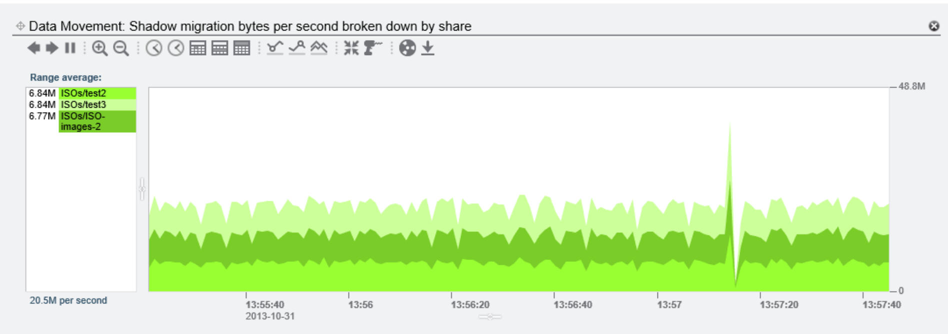

Figure 17. Shadow Migration analytics display in the Oracle ZFS Storage Appliance BUI.

Conclusion

Shadow Migration, a feature of Oracle ZFS Storage Appliance, adds another valuable tool to the administrator's toolbox. It provides an automated and measurable method to perform migration from legacy platforms to Oracle ZFS Storage Appliance and also within an Oracle ZFS Storage Appliance family appliance when necessary.

Shadow Migration is not the answer to every migration problem, as there are no standard tools that outshine every other for all sizes and types of files. Where Shadow Migration does shine is in the performance monitoring of the migration process. The advanced analytics capability of Oracle ZFS Storage Appliance allows monitoring of progress not only at the individual files level but also by share, along with the client I/O access patterns. This level of information greatly facilitates the accurate estimate of total amount of time for the migration to complete.

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

How to Migrate Data to Oracle ZFS Storage Appliance Using Shadow Migration
October 2016, Version 1.1
Andrew Ness
Oracle Application Integration Engineering