Best Practices White Paper

Dell EMC Isilon: Best Practices for Semiconductor Design Environments

Abstract

This document provides best practices for configuring and managing Dell EMC™ Isilon™ scale-out storage, powered by the OneFS™ operating system, in a semiconductor design environment.

February 2020

H18076

Revisions

Date	Description
February 2020	Initial release

Acknowledgements

Authors: Anjan Dave, Balachandran Rajendran, Bob Williamsen, Callan Fox, Jignesh Bhadaliya, John Cassidy, Timothy Wright, Yifan Zhang, Lawrence Vivolo, Anupam Pattnaik.

Table of contents

Executive summary

This document provides best practices for configuring and managing Dell EMC™ Isilon™ scale-out storage, powered by the OneFS™ operating system, in a semiconductor Integrated Circuit (IC) design environment, commonly known as Electronic Design Automation (EDA) or chip design. It includes best practices for aggregated EDA workflows, system tuning and data management strategies.

1 Introduction

The past decades have brought incredible growth and innovation in the semiconductor industry. Moore's Law can take credit for much of this innovation. Moore's Law predicts that the number of transistors on a chip will double every 2 years (mainly by making individual transistors smaller). EDA workflows, used to design semiconductor chips, have evolved to address the challenges of growing chip size and increased manufacturing complexity that has resulted. Requirements for compute (performance) and storage (total TBs) have also tracked with Moore's Law – with CPU performance and storage requirements doubling every two years.

Eight years ago, however, the laws of physics made it more difficult to shrink individual transistors – slowing down the evolution. Engineers adapted by changing focus from growing CPU performance to increasing total core count per CPU. Engineers also increased transistor count per chip by making the chips much larger. Engineers continue to find new ways to shrink transistors as well, though more slowly. The net of all this is that Moore's Law continues, but chip complexity is growing faster, which directly impacts infrastructure. Currently, CPU core requirements continue to double every 2 years. But storage requirements quadruple. This change has uncovered a new performance bottleneck – storage.

Unfortunately, existing storage technologies, while able to accommodate the acceleration in capacity requirements, they have done so at the expense of storage performance. This change has effectively shifted the EDA tool performance bottleneck – historically CPU performance – to storage performance. This problem can be traced to legacy scale-up storage architectures, which have dominated the semiconductor design market for decades, yet remained fundamentally unchanged.

Today market leading organizations are taking a holistic approach to IT infrastructure, looking for any opportunity to improve performance. Many are migrating to Dell EMC Isilon, which invented the scale-out storage architecture. They are also leveraging smarter job schedulers that can share resources more effectively. Design teams are looking to AI as a tool to give them a competitive advantage. Though most EDA flows are NFS driven, CAD and IT teams are looking to object storage to leverage the S3 access, geo replication, and distribution capabilities of Dell EMC ECS storage as well. For Moore's Law to remain alive in the new decade, organizations must innovate across the complete stack including CAD design flow, engineering infrastructure, and IT. With time-to-market requirements always a priority, the need for high performance, scalable, enterprise-class, and globally distributed design infrastructure is of utmost importance.

1.1 Dell EMC Isilon for EDA workloads

Dell EMC Isilon combines a scalable hardware platform with a parallel software architecture to optimize data storage for EDA workloads. Isilon scale-out, network-attached storage (NAS) is a fully distributed, symmetrical system that comprises clustered storage nodes. The OneFS operating system unifies the memory, I/O, CPUs, and disks of the nodes to present a single, linearly scaling file system.

Adding nodes adds capacity, performance, and resiliency to the cluster, and each node can process requests from EDA compute grid clients, while taking advantage of the entire cluster's performance. The Isilon architecture contains no single location for the data, no concept of a controller head and no RAID groups. The result is a highly efficient, scalable, and elastic architecture.

One problem with traditional, scale-up storage architectures, is that they create performance bottlenecks that deteriorate at scale. The controller is the primary bottleneck and attaching too much capacity to the controller can saturate it. Storage system bottlenecks can negatively affect wall-clock performance for concurrent jobs,

which can lengthen the time to market for a new chip. The distributed Isilon architecture eliminates the single-head CPU saturation point of a controller. A large number of concurrent jobs, often resulting in substantial amounts of metadata operations, can run without saturating the storage system, shortening the time to market.

Traditional storage systems also use disk space inefficiently. The uneven utilization of capacity across islands of storage requires manual intervention to rebalance volumes across aggregates and to migrate data to an even level — work that increases operating expenses. The inefficient utilization also negatively affects capital expenditures because extra capacity must be set aside as storage overhead.

In contrast, OneFS evenly distributes data among a cluster's nodes to maximize storage efficiency. An Isilon cluster continuously balances data across all the constituent nodes, conserving space and eliminating much of the capacity overhead that traditional storage systems require. The efficient utilization of disk space and ease of use allow Isilon storage to reduce both capital and operational expenses.

Traditional storage systems also have multiple points of management. Each filer must be individually managed, and this management overhead increases the total cost of ownership and operation expense (OpEx). The lack of centralized management also puts organizations at a strategic disadvantage because it undermines their ability to expand storage to adapt to growing datasets and fluctuating business needs, which can increase time to market. Multiple volumes of data are presented to users, which must work within the limits of each volume. The OneFS architecture, on the other hand, presents data within a single volume, regardless of the number of nodes within the cluster. With its single volume, Isilon’s OneFS architecture delivers a high return on investment by centralizing data management.

By scaling multi-dimensionally to handle the growth of data in the semiconductor industry, an Isilon cluster lets organizations adapt to fluid storage requirements, add capacity and performance in cost effective increments without disruption, and improve run times for concurrent jobs. This paper describes these benefits of Isilon storage and provides best practices for configuring and managing OneFS scale-out storage, powered by Isilon systems, in a semiconductor computer-aided design environment.

1.2 Dell EMC Unstructured Data Solutions Overview

Unstructured data does not have any pre-defined format or organization and does not reside in a standard database. Structured data is easy to search quickly using standard business analytic tools. Searching unstructured data takes different techniques and tools and is more difficult due the myriad of data sources and types that could be involved. However, unstructured data makes up a growing proportion of total data generated and accounts for over 80% of all data stored. Unstructured data requires a different approach regarding storage, particularly in the electronic design automation (EDA) field in which data is primarily unstructured.

1.2.1 Isilon

Dell EMC Isilon storage is a scale-out NAS platform designed to meet the data storage challenges of the unstructured data age. Unlike traditional scale-up storage, Isilon storage can scale to multiple petabytes of data in a single volume. Isilon single volume eliminates the need for administrators to set up and manage multiple storage silos, each with their own sets of RAID groups, aggregates, volumes, mount points, and hot spots to manage. These issues, common with legacy, scale-up storage, can make storage very difficult and expensive to manage, but these problems disappear with Isilon storage.

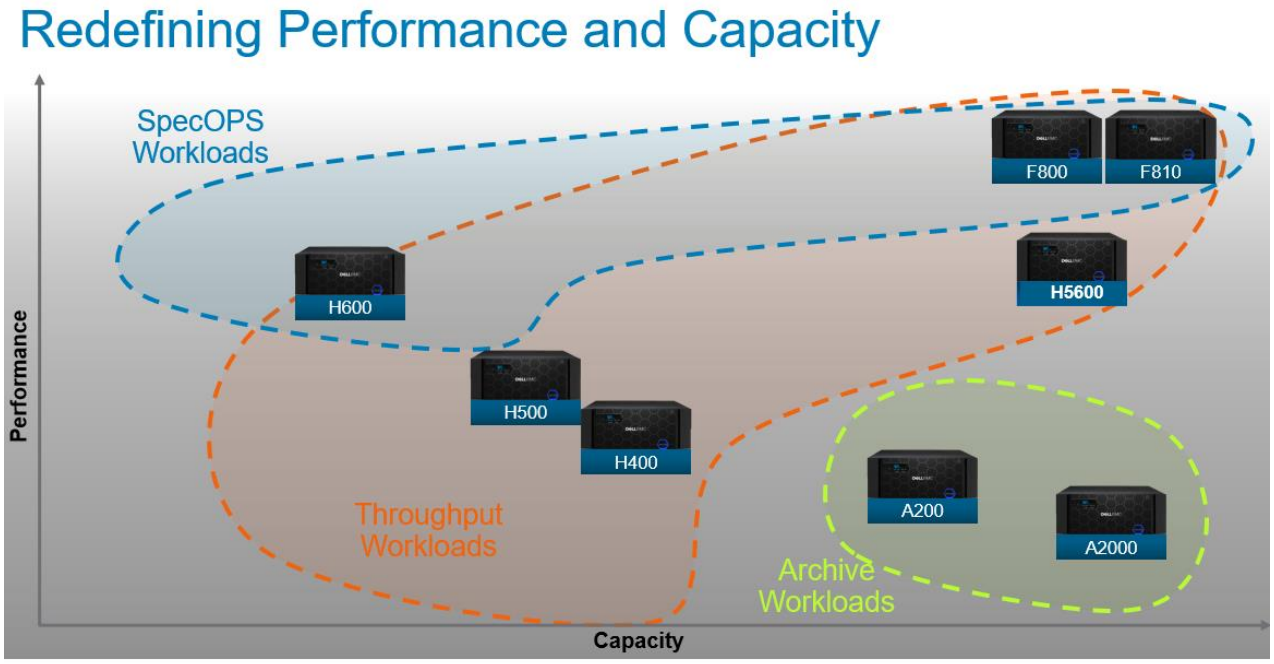

Figure 1: Dell EMC Isilon Storage Node Types by Workloads

Isilon storage offers a variety of node types that are designed for each part of a typical EDA workload. At the highest end of performance, the F800 model offers the highest performance thanks to its all-flash architecture. The F810 model adds to the performance of the F800 with inline compression and deduplication for maximum storage efficiency. For scratch space, the F800 may be the optimal choice compared to high-performance workloads that may realize more benefits from the F810 data reduction features. The Isilon family also features hybrid node types with a mix of flash and spinning discs. Though these are common with other workloads, that the vast majority of EDA work flows demand the highest performance, for which all flash storage nodes are recp,,emded. Finally, the A Series nodes offer the lowest cost per terabyte. The A200 model is designed for efficient back up and active archiving, while the A2000 model is designed for large, colder archives for long-term preservation.

1.2.2 ECS

Because of the explosion of total data requirements, less-expensive data repositories consisting of object storage (often referred to as cloud storage) have become popular. Dell EMC ECS is a third-generation object storage platform and is an ideal choice to meet this particular need. ECS is designed to be an on-prem cloud storage system, where ECS clusters can be distributed and shared across multiple geographic locations. This enables geographic data distribution and access along with geographic data protection. In addition to on-prem clouds, ECS can also be used in conjunction with public clouds such as Google Cloud Platform (GCP), Microsoft® Azure® and Amazon®. ECS has shown to be a more economical option than public clouds, especially as data egress charges increase.

ECS has three different models to match a range of needs. The EX300 entry-level model scales from 60 TB to 1.5 PB; the EX500 midrange model scales from 480 TB to 4.6 PB; and the EX3000 model, which scales from 2.7 PB to 8.6 PB, all in single 40U racks. The cluster size is unlimited.

1.2.3 SmartPools

SmartPools enables automatic migration of data up and down the different tiers within an Isilon cluster – all within a single volume. As noted earlier, different node-types can be used for each tier to provide multiple tiers based on performance/capacity requirements. Data migration policy is configured by each customer, allowing for maximum flexibility and customization. This makes it easy for an administrator to optimize the performance and cost profile for every file. A common application for SmartPools in EDA is to automate migration of project data to lower-cost storage (including archive) after a project has completed. Moving data back from lower-cost storage can also be automated. This would be useful should a project need restored due to unexpected chip revisions, or to develop a next-generation chip. Behavior is controlled by policy, can be based on last time accessed (for example), and is fully configurable by the administrator.

1.2.4 CloudPools

CloudPools is an extension to SmartPools that enables tiering of data from an Isilon to an ECS system, or to a public cloud object store, based on administrator-defined policies. This feature helps preserve the primary storage footprint by replacing old, infrequently accessed files with small reference or stub files. This frees up the storage space for newer, more frequently accessed data.

Note that most chip design workflows deal with large quantities of small files, which may be smaller than the stub files they would be replaced with. For EDA it is considered best to use traditional archive methods for infrequently accessed data. Data management and archiving are discussed later in this document.

1.2.5 ClarityNow

ClarityNow manages data from different types of storage including Isilon, ECS, public and private clouds, to provide a unified, single pane of glass file system view. Users can find, use and organize files; IT administrators can manage and monitor storage infrastructure - eliminating the problem of data silos by proving a holistic view into heterogeneous storage platforms on-prem and in the cloud.

ClarityNow can ‘tag’ an attribute and use that tag to query millions of files across any storage system, enabling business users, as well as IT, to view data in a true business context. ClarityNow enables data mobility with bi-directional movement between file and object storage. This gives organizations the ability to see their data in the right context and to optimize costs across their storage environment. Business owners are empowered to use self-service archive capabilities to move files to the most appropriate storage tier such as archive or the cloud. Industries that benefit the most are those that are dependent on technical workflows -- including EDA.

1.2.6 Cloud Storage for File

Dell EMC offers fast access to the elastic compute resources available in the public clouds. Increasingly, EDA design-houses and foundries are looking for ways to add massive amounts of CPU capacity (cores) on-demand to avoid the expense of acquiring new compute hardware and reconfiguring their on-premise compute grid for short-term needs. Dell EMC offers a multi-cloud storage service that allows shopping for the best spot pricing on compute every time when running a CPU-intensive job in the cloud. This capability makes data available to multiple public (and private) clouds simultaneously – allowing engineers to benefit from tools (AI/ML/DL, analytics, etc.) from multiple cloud providers. Multi-cloud storage also eliminates vendor lock-in as it features no egress fees to/from on-prem storage. This is a fully managed Dell EMC service powered by Dell EMC partner Faction which has data centers located close to the major public cloud providers to minimize latency.

2 EDA workflow

This section describes the overall chip design workflow to help readers understand at a high level different stages of chip design. Understanding this workflow, and the associated infrastructure for executing the workloads sets the stage for identifying challenges that undermine the performance, efficiency, and scalability requirements for chip design. Historically, the EDA workflow is described, at a very high level, as consisting of the Front-end and Back-end flows though it is not uncommon for the flow to be broken into several high-level phases as shown in the figure 3 below. Within each phase there can be many individual EDA tools used by the designers.

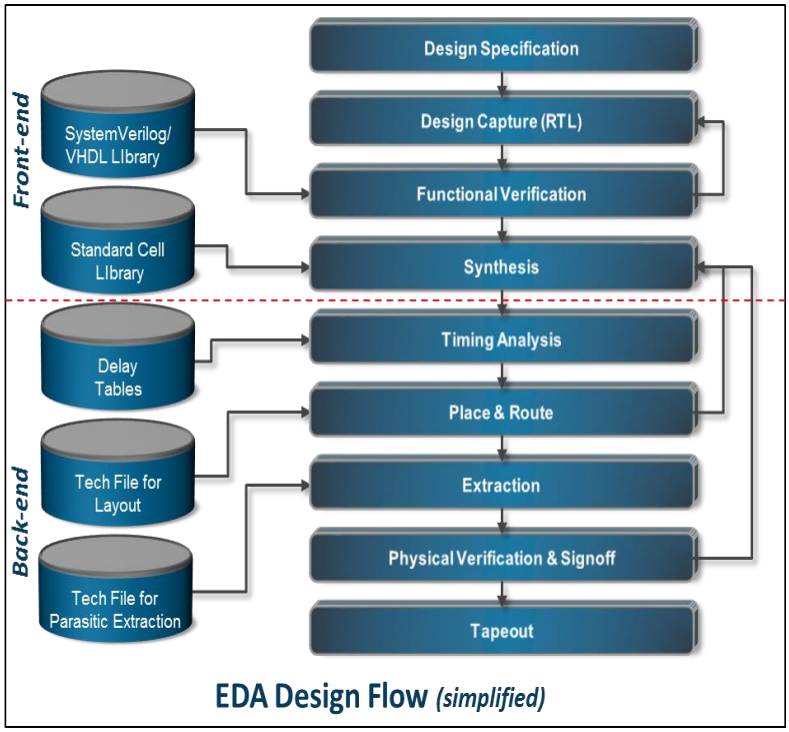

Figure 2: EDA Design Flow

2.1.1 Functional Specification, Design, Verification (Front-end)

Each new chip design starts with design specification, where engineers define the precise parameters of the design – including functional features, power consumption, performance levels, physical chip size and more. Once the scope of the project is defined, engineers then build high-level architectural models of the design to help define the final architecture, verify that it meets performance goals, and begin building the verification environment. The design is then implemented using a specialized language (Register Transfer Language, or RTL) such as SystemVerilog or VHDL to capture the design in a format that can be synthesized to gates (groups of transistors). Transistors are one of the building blocks of semiconductor design. Typically, there is some overlap between high level modeling and RTL capture. Design verification at this stage is considered “functional verification” and occurs throughout the design cycle, including before and after design synthesis. Functional verification (simulation) checks that the design, as coded and synthesized, behaves as expected. (Ex: 1 + 1 = 2, not 3). For a complex design, hundreds of thousands of concurrent simulation “jobs” are typically run to save time. Functional verification does not check for physical attributes, such as heat dissipation, voltage variations and other electrical challenges. These are addressed in the back-end part of the flow.

Efficiency in creating, scheduling, and executing build and simulation jobs can reduce the time it takes to bring a chip to market. There are workflows used in design specification phase that generates an I/O-intensive workload when many jobs run in parallel: EDA applications read and compile millions of small source files to build and simulate a chip design. A storage system manages the various design projects and files so that different users, scripts, and applications can access the data. During the design verification stages, the data access pattern tends to be random, with many small files. Some of the workload requires high levels of concurrency because of the large number of jobs that need to run in parallel, generating a random I/O pattern.

2.1.2 Physical Design (Back-end)

During the back-end design phases, the data access pattern becomes more sequential. Some of the physical design implementation workflows for block level design tends to have a smaller number of jobs with a sequential I/O pattern that run for a longer period of time. The output of all the jobs involved in a chip's design phases can produce terabytes of data. Even though the output is often considered transient, and placed on scratch space, the data still requires the highest tier of storage for performance. Within the storage system, workflows tend to store a large number of files in a single directory, typically per design phase, amid a deep directory structure on a large storage system. The performance-sensitive project directories, including those for both scratch and non-scratch directories, dominate the file system and can become a bottleneck for legacy, scale-up storage solutions.

The directories contain source code trees, RTL files that define logic in Hardware Description Language (HDL), binary compiled files after synthesis against foundry libraries, and the output of functional verifications and other simulations. Typically, the RTL project directories that contain source code are minimal in size, while the project directories used for simulations dominate the overall capacity utilization of the storage system.

3 System tuning

This section introduces several different configurable settings that are applicable to chip-design workflows, based on the collective experience of Isilon customers, field and engineering resources. One may wonder why Isilon is not configured with the appropriate settings out of the box. Isilon is a scale-out clustered NAS system that caters to a wide variety of workflows in many different industries. The operating and performance requirements for specific industries like semiconductor, media and entertainment, healthcare, and others are quite different, and in some cases are contradictory in nature. For example, while the media and entertainment industry tends to have serialized workloads focused on larger files, semiconductor computer-aided design workflows are metadata intensive, and have a mix of mostly small and large file sizes. The flexibility of Isilon OneFS allows it to be tuned for specific, predominant workflows, and together with the family of different node types available, a cluster can be configured to deliver optimized performance throughout the EDA workflow.

The list of configurable settings in this section is comprehensive as of the writing of this paper, presenting many options to help aid the decision of which tunable makes more sense for the environment. However, with each new version of OneFS, certain settings may become obsolete or redundant in nature (and could default to the same value that was set manually), or some new parameter may be introduced based on customer feedback.

Some settings may represent a tradeoff, such as space efficiency over performance, or require adhering to business needs such as prioritizing data protection at the expense of some performance. One should first gain a good understanding of the tunable and weigh its effect on the workloads on the cluster before making changes. It is highly recommended to test the settings in an isolated environment and engage with account resources for recommendations and/or any questions.

Semiconductor chip design workflows have dependencies on the entire infrastructure including: compute client type (including CPU/memory), OS settings, EDA tool versions, mount options, networking configuration, authentication providers (NIS, LDAP, AD), name services, time servers, and much more. When framing any test, it is best practice to take a holistic approach that accounts for these environment variables and nature of the workflows hosted on the cluster. For example, the number of hard links on a cluster is a configurable setting, but if the need arises for a much higher number of hard links, careful evaluation of the workflow should be done prior to making changes to fully understand why this change is needed. Similarly, using either a hard or soft quota would work, but the recommendation for chip design environment is to use soft quotas configured with a short grace period given the general nature of EDA workflows.

Dell EMC also recommends that a test environment be maintained that includes all client OS versions in use. Having the ability to test various EDA tool versions, and even testing a new shell version in an isolated environment, can be helpful in debugging issues.

The OneFS configuration settings listed in the following subsections are recommended. Note: The settings presented have been divided into three levels: basic, advanced, and expert. The expert level settings should be performed only with the help of Isilon support team personnel. The Dell EMC systems engineer (SE) assigned to the account should be consulted first to ensure applicability and validation of the settings for the specific customer environment and workflows. Additionally, if you need assistance with managing the sysctls, refer to this support article: https://emcservice--c.visualforce.com/apex/KB_BreakFix_clone?id=kA2j0000000R4oy&lang=en_US&pubstatus=o. The sysctl based examples provided throughout the document should be applied in accordance with the instructions presented in this article.

3.1 Basic settings

3.1.1 Tunable placement of /ifs/.ifsvar

The /ifs/.ifsvar directory contains many OneFS configuration files and internal data structures. This part of the file system is placed on one node pool at a time, and its placement is dictated by the system node pool which is by default the first node pool in a cluster. Also, the Job Engine checkpoint files, auditing files, and other files are part of this area. If the cluster is homogeneous in nature (all nodes of same type), this tunable is not applicable.

The performance of /ifs/.ifsvar can greatly impact the cluster's overall performance. Significant performance improvements have been seen through placing /ifs/.ifsvar on the faster nodes, and in particular, all flash nodes.

3.1.1.1 Default settings

By default, /ifs/.ifsvar is placed on the first node pool that is commissioned.

3.1.1.2 Recommended settings

It is recommended that when implementing a new cluster, the faster nodes are deployed first. However, where a cluster already exists, the /ifs/.ifsvar directory should be moved to the highest performing node pool.

Note: The highest performing node pool may also be the busiest node pool, so check the system load before changing its placement.

Determine where /ifs.ifsvar is located, with any file in the /ifs/.ifsvar directory:

```
isi get -DD  /ifs/.ifsvar/isi_license.xml | grep pools
```

Look to see what nodepool it is assigned to, which is where the contents of .ifsvar are located.
Identify what nodepool(s) are configured as **System**. This can be multiple pools.

```
disi -I diskpools ls -v
```

Look for the nodepools that have an **S** in the flag column. These are system nodepools.
Based on the output of the prior command, you may need to remove or add the system flag.

3.1.1.3 Update the tunable

Define the system pool desired (where .ifsvar will move to).
Turn off automatic management.

```
isi_gconfig smartpools.diskpools.manually_manage_system_flags=true
```

Assign the system pool. This example removes the system flag from the A200 disk pool.

```
disi -I diskpools modify --force --system=false a40_200tb_800gb-ssd_16gb
```

Switch automatic management back to false.

```
isi_gconfig smartpools.diskpools.manually_manage_system_flags=false
```

The next time Smartpools runs, the .ifsvar directory moves to the **System** nodepool.

Note: If SmartPools is not licensed, the directory moves the next time the **SetProtectPlus** job runs.

Validating this tunable can be difficult because the response time to this part of the filesystem is not measured with standard tools. Running isi statistics heat shows that that Isilon storage performs a massive amount of IO on /ifs/.ifsvar. The response time of these operations would improve in faster nodes and should improve cluster background system performance.

3.1.2 Access time tuning

When a file is updated on a file system, the access time is updated. OneFS allows configuration on the granularity of **access time** updates. Changing this setting to 1 day means that access time will only be updated if the file was accessed more than one day ago.

3.1.2.1 Effects on OneFS

Access time means that a read operation on metadata turns into a write as well since OneFS must update the inode. The result is increased CPU and disk traffic on a given system. This effect can be mitigated by the use of metadata acceleration. In either case, the extra workload is not necessary if the workflow does not need this level of granularity.

3.1.2.2 Default settings

By default, this is set to 1 day.

3.1.2.3 Recommended settings

Access time should be set relative to the customer's workflow requirement, erring on the side of a larger duration. In general, anything less than one day is not required since SmartPools does not run more than once per day by default.

Use the standard option in the WebUI to check this tunable.

3.1.2.4 Update the tunable

Use the standard option in the WebUI to update this tunable.

This metric that can be difficult to validate. The effect on the system results in less filesystem updates since metadata reads no longer need a metadata update. There may be a noticeable effect when observing disk activity before and after the change, but there are many attributes that affect the impact:

- How much the time granularity changed. If it was reduced from 1 second to 1 day, there should be a large change.
- The amount of metadata read operations. These operations no longer need a disk write. The easiest way to look at this is to use InsightIQ to break down the operations by type.

3.1.3 Metadata inode sizing and force_8k_inodes

This is not really a tunable, rather it's a setting that may address a sizing issue in specific use cases.

Generation 6 storage pools by default are configured to use 8K sized inodes. Generation 5 and earlier storage pools default to 512b inodes.

Migrating data from a previous generation node type to a generation 6 node with SmartPools will result in a significant increase in SSD capacity utilization on storage pools with SSDs configured for metadata acceleration. The issue is exacerbated if metadata read/write acceleration is configured.

Generation 6 nodes can be reverted back to 512b inodes to reclaim capacity on the SSDs.

Metadata describes information about data stored on OneFS. Data is stored within files and folders, and these objects are tracked by metadata. Unix-based systems refer to the basic building block of metadata as an inode. Without inodes, the data stored on the data drives of a cluster are useless, and metadata is mirrored in OneFS.

The mirroring of this metadata is controlled by the protection requirements of the cluster. For example, +2d:1n needs 3 copies to provide protection against 2 failures, so metadata is mirrored thrice. All metadata in a chain needs to be protected, and directories by default are also mirrored at 1 level higher than a typical file. This is configurable.

In OneFS, the minimum inode size is 0.5 KiB, and can be extended larger in the following instances:

- The file exceeds approximately 10 MiB in size
- Many security permissions are set

In some cases depending on the drive type (usually drives less than 6TB) used in the node, the inode size can change to 8k and this can significantly change the space consumption of a system. This is especially true in large-file workloads like EDA, and some media and entertainment workloads.

There is a force_8k_inodes parameter, and if it is true, inodes are forced to 8k. This issue is expected to be corrected in OneFS versions 8.1.0, 8.1.2, and 8.2.

3.1.3.1 Effects on OneFS

Increasing an inode from 0.5 KiB to 8 KiB could result in an additional disk space consumption of metadata especially with workloads consisting of predominantly small files.

3.1.3.2 Default settings

This setting may be enabled on Gen6 nodes accidently depending on OneFS version.

3.1.3.3 Recommended Settings

Run the following command to check if the output contains an “I” in the flags column. If it does, open a service request to see if the forcing of 8k inodes is an applicable setting for your environment. You can mention kb529696 to support personnel (it is a restricted KB).

```
#disi -I diskpools list -v
```

Note: All Isilon F800 and F810 models are hardcoded to 8k inodes, no changes needed for these nodes.

Have the service request owner refer to the Dell EMC Support knowledge base article at https://support.emc.com/kb/529696.

3.1.3.4 Update the tunable

See above description.

3.1.4 System BTREE placement

A BTREE is a self-balancing tree data structure that maintains sorted data and allows searches, sequential access, insertions, and deletions in logarithmic time. OneFS uses BTREEs for many internal data structures, and these are mirrored like metadata entries.

3.1.4.1 Effects on OneFS

The placement of the system BTREE can significantly increase performance depending on the use of HDDs or SSDs. The placement of the BTREEs can be controlled.

3.1.4.2 Default settings

For systems that have SSDs, the default setting is to have one of the mirrors on SSD. This is very similar to metadata read acceleration.

3.1.4.3 Recommended settings

This setting is recommended when the cluster has SSDs for metadata acceleration (Read Only (RO) or Read Write (RW)) and there is enough space. Put all mirrors on SSD. The possible settings are 1 or 3 (which are enumerators), and a setting of 1 means partial on SSD, while 3 means all mirrors on SSD. This setting deflects I/O from disks to SSDs. The system BTREE is a destination for many internal-filesystem, metadata-tracking-related operations. This is a filesystem within a filesystem. STF, STB, and ADS blocks are in system BTREE.

3.1.4.4 Check the tunable

There are three mirrors to check – System Btree mirrors, QAB mirrors, and System Delta mirrors. Check the number of system BTREE mirrors with:

```
sysctl efs.bam.layout.ssd.sys_btree.mirrors
```

Check the location of system delta blocks or inodes (differences in inode changes) mirrors:

```
sysctl efs.bam.layout.ssd.sys_delta.mirrors
```

There are delta blocks of filesystems for STF blocks. The deflection in STF and STB block I/Os can comprise up to 5–10% that is diverted to SSD. This could equal roughly 200 I/Os per SSD deflected from spinning disks. ShadowStores also use this. This setting is not applicable for F-series nodes.

Use caution as to not to fill up SSDs. For example, if SSDs are 60% full, do not enable this setting. In general, for SSD sizing, if the cluster has 75% SSD utilization it is beginning to be undersized.

Check the location of QUOTA allocation blocks:

```
sysctl efs.bam.layout.ssd.qab.mirrors
```

As for the Quota Allocation Blocks (QAB) mirrors, if not using quotas, there may not be a noticeable difference. A QAB is a block that tracks quota rules and is written to disks. This is serialized to the journal, and that could block other operations. These blocks are assigned some nodes, so whenever a quoted directory changes, the computational changes are sent to those nodes that have the QABs (I/Os). With many quotas and active systems, those nodes can be busy when doing I/Os for quota computations. Putting them on SSDs would be recommended. They need to be written out as fast as possible to prevent spending more time serializing the journal.

Update the three mirrors:

The sysctl command can be used to update the mirrors. In addition, in OneFS 8.1.x onwards, this can be performed with the command **isi storagepool settings modify** command. **--ssd-system-btree-mirrors=all.**

The below sysctl –d command shows the description of these settings:

```
sysctl -d efs.bam.layout.ssd.sys_btree.mirrors
efs.bam.layout.ssd.sys_btree.mirrors: Layout: number of metadata mirrors to
place on SSDs:
0: Disabled; treat SSDs as any other drive
1: One mirror on SSD
2: Invalid (previously all but one mirror on SSD)
3: All mirrors on SSD
4: No mirrors on SSD
```

Below sysctl commands would modify the values of all three ssd mirror types to 3:

```
sysctl efs.bam.layout.ssd.sys_btree.mirrors=3
sysctl efs.bam.layout.ssd.sys_delta.mirrors=3
efs.bam.layout.ssd.qab.mirrors=3
```

To make this persistent, update the file /etc/mcp/override/sysctl.conf.

To modify these settings using the “isi storagepool” command do the following:

```
isi storagepool settings modify --ssd-system-btree-mirrors=all

isi storagepool settings modify --ssd-qab-mirrors=all

isi storagepool settings modify --ssd-system-delta-mirrors=all
```

In terms of validation of these settings, this tunable would take workload away from the data drives and onto the SSD. However, there is not an easy way to measure the response time to these data structures, so there is no justifiable need to validate this.

Note: If using L3 cache, these settings can be disregarded since L3 cache formats SSDs for exclusive usage. If the above settings are changed, run the SmartPools job after making the changes.

3.1.5 SSD metadata strategy settings

Metadata performance is crucial to a filesystem and thus to OneFS performance. OneFS has configuration options to control the placement of metadata between HDDs (default) and SSDs.

3.1.5.1 Effects on OneFS

The primary effect on OneFS is performance. Moving metadata to SSDs has a joint effect of better performance for the metadata and freeing up the drives from small random reads (and writes in metadata update cases). Metadata performance often affects the user experience in response time (fast responsiveness and folder browse speed).

Note: The settings in the file explorer or filepool policy for metadata acceleration are applied secondary to the node pool setting. If the pool is set to L3, even if the file pool policy is set to metadata RO, it will be ignored.

3.1.5.2 Default settings

By default, any system with SSDs (all Gen 6) will set metadata performance to L3. All node types except A200/A2000 can be changed from L3 to either metadata RO or RW

Note: The A200 and A2000 models only cache metadata by default.

3.1.5.3 Recommended settings

L3 is a persistent cache for data and metadata, and writes are not issued into L3. L3 only caches data reads of non-continuous blocks (not sequential) of less than 128 KB. In most cases, this is not substantial, but this does depend on the workload.

L3 metadata performance is generally less than that of metadata RO since metadata is not resident in SSD, and any changes need to be written to the drives. Tests in SpecSFS show a small gap, but this is mostly due to cache re-hit. However, in a real-world environment, the re-hit rate may be high, which can be measured with the L3 cache hit (split out by data/metadata).

In most cases, metadata RO and RW (preferred) would be optimal. If desiring a measured result, and the system is currently using L3, examine the cache stats for L3 broken down by data and metadata.

- Moving to metadata RO or RW makes all metadata hits 100% since they are always available on SSDs.
- This setting helps with data drive usage since the SSDs take the metadata workload.
- While data drives are now freed from metadata workload, the L3 data hit percentage mostly returns to the data drives. If there is a workload that uses this heavily, this setting is not recommended.

It is recommended to use metadata RW where possible, allowing for proper sizing.

Another recommendation is to maintain the SSD utilization for active datasets. Use a filepool policy that specifies if files are not modified in *X* amount of time, the SSD strategy is set to metadata read. This ensures the SSD space is used up for active data only. In addition, for snapshots, set this to metadata read or none.

3.1.5.4 Update the tunable

This can be done via the GUI. Go to File System -> Storage Pools -> Smart Pools tab. Click on View/Edit for the concerned nodepool to uncheck the box for L3 cache. Following this, go to the File Pool Policies tab and modify the default filepool policy and any other user-defined policies to set the appropriate meta-data strategy.

When changing from L3 to metadata read/write or read, there are two considerations for validation:

- If the response times for metadata operations are improved
- If the application or user experience has improved
- If there is an impact to read operations since L3 is no longer present to cache small non-continuous reads smaller than 128K

Note: In EDA workflows, if down-tiering data to an H500 or H400 pool, consider using metadata read on that tier. Even if not down-tiering, a policy can be created to do metadata reads for data that has not been accessed in a long time. This helps preserve available SSD space.

3.1.6 Job Engine and Snapshot Settings

There are three settings presented here that may be worth considering. First two are related to job engine, and the third is about snapshots. Please note, these settings are dependent on the OneFS versions. We present them here since the audience for EDA comprises of several different OneFS versions.

3.1.6.1 Effects on OneFS

These settings are presented here to help reduce the impact of certain jobs, thereby managing the cluster resources in favor of user workloads.

3.1.6.2 Default settings

See the Recommended Settings section below

3.1.6.3 Recommended settings

1. The job engine should start LIN-based jobs, as shown in the following setting. (**Auto** does not perform this setting.)

```
# isi_gconfig -t job-config jobs.common.lin_based_jobs
jobs.common.lin_based_jobs (enum enable_state) = Auto
# isi_gconfig -t job-config jobs.common.lin_based_jobs=true
# isi_gconfig -t job-config jobs.common.lin_based_jobs
jobs.common.lin_based_jobs (enum enable_state) = True
```

2. Job engine settings: Prior to OneFS 8.2, disable multiscan, and collect. Note, Multiscan is a combination of two jobs, AutoBalance, and Collect. AutoBalance, along with the Collect job, is run after any cluster group change, unless there are any storage nodes in a “down” state. Mediascan can be run at a lower frequency than the default once a month.

Also, change all jobs’ impact setting to LOW. Setting jobs to MEDIUM does not realize much gain and can complicate administration.

3. EDA design workflows use could benefit from using an EDA_OFF_HOURS schedule based on the typical **off hours for your specific environment**. This can be done by creating a new schedule under the Job Operations -> Impact Policies and changing flexprotect, SmartPools, and impactful jobs to run during this schedule only. Once this is complete, the number of workers assigned to these jobs can be increased.

4. Prior to OneFS 8.2, for EDA workflows that use snapshots, change the sysctl value for max_active_snapids (see below) to **5**. This essentially directs the system to locate more than one active snapshot during the snapshot deletion, consuming more resources. Unless the cluster is a new installation on OneFS 8.2 or higher, this sysctl should be set to 5.

```
# sysctl -d efs.snapshot.max_active_snapids
efs.snapshot.max_active_snapids: max non-deleted snapids to search while pruning
governance list
# sysctl -a efs.snapshot.max_active_snapids
efs.snapshot.max_active_snapids: 1
```

To change this tunable to 5, do the following:

```
# sysctl efs.snapshot.max_active_snapids=5
```

3.1.6.4 Update the tunable

Refer to the Recommended Settings section.

3.1.7 Endurant cache setting

Endurant cache (EC) was introduced in OneFS 7.1.1 and was aimed at reducing the latency of synchronous (immediate or stable write) small block random writes. The primary goal of this feature was to help with latency of VMware-type workloads on Isilon storage.

3.1.7.1 Effects on OneFS

By default, stable storage is written to disk, and a synchronous write requires the storage system to write to stable storage. Because the write is smaller than the Reed Solomon stripe, this causes a READ/WRITE/MODIFY penalty and impacts response time.

EC effectively works around this process by writing the small write to a small part on the journal that is protected. EC then has to de-stage these writes to disk over time. If multiple writes are issued to the same stripe, the de-staging of these writes is more efficient since the R/W/M penalty is only paid once.

An issue occurs when the EC space is not large enough. While EC helps with bursts, continuous traffic of small-block writes may cause it to constantly catch up, and this can result in lower performance compared to not using EC.

3.1.7.2 Default settings

EC can be managed on and off globally, and tuned on or off per directory.

3.1.7.3 Recommended settings

The recommendation is to turn this off for EDA globally.

3.1.7.4 Update the tunable

To disable globally, add this to /etc/mcp/override/sysctl.conf:

```
efs.bam.ec.mode=0
```

3.1.8 The ‘anywhere’ setting in SmartPools

OneFS offers data tiering, but sometimes this is implemented without proper planning and can lead to issues.

File pool policies rules are created to control the placement of data. Rules are processed in order until a rule matches. Otherwise, the default rule is applied.

3.1.8.1 Effects on OneFS

When the destination pool is set to **anywhere**, the data storage is written to any node pool in an even manner. For example, if there are two pools, H500 and A200, anywhere will spread an even percentage of data to both.

If another rule sends 200 TB of data to the A200 pool, making it 40% full, the anywhere rule sends more data to the H500 pool until the two tiers are even at 40%, and then returns to even distribution. This creates unpredictable performance profiles and poor customer experiences.

3.1.8.2 Default settings

The default setting is to use the destination node pool as **anywhere.**

3.1.8.3 Recommended settings

- Group all like-performance node pools into a tier, for example, X410 and H500, or two different pools of H500 with different drive sizes.
- Set the highest performing node pool or tier as the default location.
- Tier as needed based on age as dictated by the workload or the reason the lower performance node pool was used.

Use the standard option in the WebUI to check this tunable.

3.1.8.4 Update the tunable

Use the standard option in the WebUI to check this tunable.

3.1.9 F800/810 settings in multi-tier cluster

Isilon all-flash nodes (F800 and F810) need to be configured to store data on SSDs. This is not a consideration for a single-tier cluster, but only for a multi-tier cluster.

3.1.9.1 Effects on OneFS

Data is not placed on the all-flash nodes unless filepool policies are configured to place data on the SSDs in the metadata configuration. An inappropriate configuration impacts the nodepool capacity and performance.

3.1.9.2 Default settings

The default configuration for all nodepools is L3. However, once L3 is disabled, the default is Metadata Read.

3.1.9.3 Recommended settings

Configure the default filepool policy to place data on the all-flash tier, and configure the metadata policy for Metadata and Data. Follow the previous recommendation to move /ifs/.ifsvar to the all-flash tier.

Use the standard option in the WebUI to check this tunable.

3.1.9.4 Update the tunable

Use the standard option in the WebUI to check this tunable.

Verify that data and metadata are now being populated onto the F800/F810 node as desired. The usage appears to increase as data is written. In a multi-tier system, a SmartPools job may need to be run, or more data needs to be written.

3.1.10 LIN-based jobs

A logical inode (LIN) is a piece of metadata. Some OneFS system jobs have two types: one that processes at the block level, and one that processes at the metadata level. There is a setting that allows the job engine to force one level or the other.

3.1.10.1 Effects on OneFS

The effect on running jobs can be substantial, and the difference between an AutoBalance and a AutoBalanceLin on a system with SSDs can be large as well. The time it takes to run some jobs can change from multiple days to a single day or less.

3.1.10.2 Default settings

By default, the job engine picks which job type is used, though this is not always optimal.

3.1.10.3 Recommended settings

For systems with metadata acceleration (RO or RW, and not L3), use LIN-based jobs for optimal performance. Some jobs may be faster using LIN-based with L3 acceleration. This is because the job is changing its processing method from block-based to metadata-based. L2 and L3 hit rates for metadata may still be high enough to warrant the change.

Check the current setting:

```
isi_gconfig -t job-config jobs.common.lin_based_jobs
```

3.1.10.4 Updating the tunable

Update the tunable from AUTO to TRUE:

```
isi_gconfig -t job-config jobs.common.lin_based_jobs=true
```

3.1.11 SMB continuous availability

SMB continuous availability (CA) or transparent failover improves the failover experience for clients that support it. At a high level, this is done with extensions to the SMB protocol and the introduction of a witness.

OneFS 8.0+ supports SMB CA and it is enabled at a share level. This is applicable for OneFS in an SMB 3.0 client environment.

3.1.11.1 Effects on OneFS

When CA is enabled, there is an additional level of tracking that is required on the back-end of the Isilon system. This is needed to ensure that the state of the connection is always up to date on the secondary nodes available for failover to them using CA.

CA causes a performance degradation since all the nodes participating in the CA process must have a synchronous state. Keeping this state has an impact on latency, and this has an effect on performance.

Generally, the faster the machine, the greater the impact. F800 node performance may be impacted by 30–40%, while an A200 node may only be impacted by 15%.

3.1.11.2 Default settings

By default, SMB CA is disabled.

3.1.11.3 Recommended settings

Only enable SMB CA for workloads that need it and if the client supports it.

In the future, the performance impact may be reduced, allowing this to become a more widely used protocol.

Note: If using CA, disable EC to prevent extreme performance degradation.

Use the WebUI to check this tunable or use “isi smb shares view <sharename>”

3.1.11.4 Update the tunable

Within OneFS GUI, go to Protocols -> Windows Sharing (SMB), and chose a share to edit, and check the box for “Enable continuous availability for share”. For additional information refer to the latest OneFS Web or CLI Administration Guide.

3.1.12 32-bit file handle

NFS uses a structure called a file handle to uniquely identify exported files. When a client requests to access a file, the server constructs a file handle that identifies it. From this point on, this identifier is used in all communications between the server and the client to access that specific file. Some EDA tools especially need 32-bit file handles returned rather than the default 64-bit file handles.

3.1.12.1 Effects on OneFS

With this setting at it's default, some EDA applications may produce an error upon execution.

3.1.12.2 Default settings

This tunable is disabled by default.

3.1.12.3 Recommended settings

Enable this tunable for EDA and any other use case that requires this setting. This can be enabled globally, or by each export.

Click Protocols > Export Settings > Advanced Default Export Settings > Client Compatibility Settings > Return 32 bit File IDs.

3.1.12.4 Updating the tunable

Click Protocols > Export Settings > Advanced Default Export Settings > Client Compatibility Settings > Return 32 bit File IDs.

Set Custom to Yes. The default setting is No, but this setting depends on the specific environment.

To enable this setting using the CLI, run the following

```
isi nfs settings export modify --return-32bit-file-ids=yes --zone=System.
```

3.1.13 NFS continuous availability

NFSv4 is a stateful protocol like SMB. In OneFS 8.0, a feature was introduced to allow NFS continuous failover.

3.1.13.1 Effects on OneFS

When an NFSv4 workload connects to a dynamic IP pool, the NFS CA feature is automatically enabled. To maintain the session state, much like SMB CA, there is a performance impact. This is due to the statefulness of the protocol which entails the cluster maintaining the state of each connection at all times. The performance impacts depend upon workloads. Some workloads may make calls that have dependency on the state of the connection.

3.1.13.2 Default settings

By default, NFS CA is enabled when a NFSv4 session connects to a dynamic IP pool.

3.1.13.3 Recommended settings

There is no specific recommendation here other than the preference of using nfsv3. If nfsv4 is required, then static Vs dynamic pool decision relies on the client behavior during the unavailability of a node.

Note: If using NFS CA, disable EC to prevent extreme performance degradation.

3.1.13.4 Update the tunable

There is no tunable here. However, you could use static SmartConnect pool for nfsv4 clients which avoids the performance impact.

3.1.14 Number of active clients

By default, OneFS shows up to 256 active clients when using isi statistics command or in IIQ. In EDA environments, due to large compute farm grid sizes, often more than 256 clients connect to each node. To accurately reflect this number, change the following parameter.

3.1.14.1 Default settings

By default, only 256 active clients are shown.

3.1.14.2 Recommended Settings

Change the configuration to check for current setting and if necessary change it allow 1024 active clients (maximum) to be reported by OneFS for NFS or SMB protocols.

```
sysctl isi.stats.client.<protocol>.max_clients
```

3.1.14.3 Update the tunable

```
sysctl isi.stats.client.<protocol>.max_clients=1024 (where protocol is nfs,
nfs4, smb2)
```

3.2 Advanced settings

The settings in this section should be modified in consultation with an SE as well as support personnel. EDA workloads can be varied in nature from one chip design center to another, and every tunable may not be applicable to each scenario.

3.2.1 Prefetch tuning

OneFS is a content-based storage system, and prefetch is performed at a logical file level, not a block level. Prefetch allows the system to fetch data ahead of the client request, in an efficient way. This improves performance in majority of cases. There are different types of prefetch algorithms and tunable settings related to them.

3.2.1.1 Effects on OneFS

With prefetch, OneFS performs reads before the client has requested them but is likely to request. Wasted prefetch operations (that are not used), is a burden on the system. Too many prefetch operations can cause unnecessary disk utilization, resulting in poor workload performance. Currently, there is not an effective way to monitor missed prefetch data (data fetched into memory that is not used).

3.2.1.2 Default settings

- By default in OneFS 8.x, the concurrency algorithm is implemented, which uses an adaptive prefetch window to fetch data dynamically.
- The random algorithm uses no prefetch.
- The sequential algorithm changes the disk layout slightly to spread stripes across more drives in a disk pool while increasing the amount of prefetch.

3.2.1.3 Recommended settings

- In general, for EDA, use the concurrency setting. Some media streaming workloads may require streaming or file name prefetch.
- Only use the random setting when there is no sequential access into that file.
- Where there are a lot of threads, the concurrency algorithm can initially fetch too many blocks which increases drive utilization. This setting can be tuned down and has been seen to reduce disk contention.
- Before modifying the tunable given below for the prefetch size, please open a service request and work with support personnel.

Check the current setting for concurrency:

```
isi_for_array sysctl isi.access.default.prefetch.l2_window_blocks.
```

3.2.1.4 Update the tunable

Add this setting to the /etc/mcp/override/sysctl.conf file.

For example, the following reduces the OneFS default prefetch from 8192 blocks (default) to 512 blocks.

```
Sysctl isi.access.default.prefetch.l2_window_blocks=512
```

This tunable can only be validated in a system that needs it. Validation will cause disk activity will drop, but this should not affect the workload. To view this disk activity, find the average of the pending I/O operations and the average time spent in the queue.

3.2.2 QAC ratio

This setting adjusts the quota allocation constituents. This setting is applicable to environments that have quotas enabled.

3.2.2.1 Effects on OneFS

If quotas are configured to enforce thresholds (hard or soft) as opposed to just reporting, then this tunable should be adjusted. EDA environments tend to rely on the enforced quota functionality to allocate disk space to end users, hence this setting is discussed here. For more information on QAC or Quota Allocation Blocks (QAB), refer to: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h10575-wp-quota-management-with-smartquotas.pdf

3.2.2.2 Default settings

```
Sysctl efs.quota.reorganize.qac_ratio 1
```

3.2.2.3 Recommended settings

The value of **8** is recommended for this setting.

```
Sysctl efs.quota.reorganize.qac_ratio
```

3.2.2.4 Update the tunable

```
Sysctl efs.quota.reorganize.qac_ratio 8
```

3.2.3 Rename event coherency

This setting adjusts the rename event coherency.

See the following KB article on cluster-wide deadlock and hangdumps while attempting lock upgrade to exclusive during rename operation: https://support.emc.com/kb/491579

If running OneFS version 7.2.1.4, 8.0.0.2, 8.0.1.1 or higher, set the following to turn off the coherency:

```
sysctl: # isi_sysctl_cluster efs.bam.rename_event_coherency=0
```

3.2.3.1 Effects on OneFS

Refer to the KB above for detail

3.2.3.2 Default settings

```
Sysctl efs.bam.rename_event_coherency 1
```

Note: the default may be a different value based on the OneFS version.

3.2.3.3 Recommended settings

The value of **0** is recommended for this setting.

To check the current settings:

Sysctl efs.bam.rename_event_coherency

3.2.3.4 Update the tunable

```
Sysctl efs.bam.rename_event_coherency 0
```

3.2.4 Soft quota container

This setting is relevant when using soft quotas and being able to see the quota as configured and not seeing the capacity of the cluster as quoted space. However, this setting makes sense when soft quota enforcement is setup with a small grace period.

3.2.4.1 Effects on OneFS

Under very heavy load where the write workload is consistently hitting the quota limits, using hard quotas may impact cluster performance.

3.2.4.2 Default settings

Use the below command to check the current value:

```
Sysctl efs.quota.soft_containers
```

Note: The default may be a different value based on the OneFS version.

3.2.4.3 Recommended settings

The value of **1** is recommended for this setting. In conjunction with this setting, the recommended way to setup quotas is to use soft quotas with a short grace period of a few seconds instead of hard quotas.

```
Sysctl efs.quota.soft_containers=1
```

3.2.4.4 Update the tunable

```
Sysctl efs.quota.soft_containers=1
```

3.2.5 Coalescer multiwriter

This is a performance tuning option. EDA workloads tend to be more concurrent and not single-streamed. This setting dictates performance improvements for single-streamed workloads but may have a small performance penalty for EDA workloads. We recommend disabling this setting.

3.2.5.1 Effects on OneFS

NA

3.2.5.2 Default settings

To check the current setting, run the following command:

```
isi_sysctl_cluster efs.bam.coalescer.multiwriter
```

3.2.5.3 Recommended settings

The value of **0** is recommended for this setting.

```
isi_sysctl_cluster efs.bam.coalescer.multiwriter=0
```

3.2.5.4 Update the tunable

```
isi_sysctl_cluster efs.bam.coalescer.multiwriter=0
```

3.2.6 SmartConnect settings

This is applicable to nfsv3 environments, which comprises of most EDA workflows. This setting affects the nfs client-server behavior for certain non-idempotent filesystem operations where the server's allocation of a delay or additional time allows for a graceful completion of an in-flight system call. This is not a tunable, but rather a configuration setting that should be applicable to all versions of OneFS.

3.2.6.1 Effects on OneFS

See above description

3.2.6.2 Default settings

```
isi network external modify --sc-rebalance-delay
isi network pools modify groupnet0.subnet1.pool0 –sc-auto-unsuspend-delay
```

3.2.6.3 Recommended settings

The values recommended are **1** and **2** respectively for the rebalance-delay and unsyspend-delay settings above. To check the values:

```
isi network external modify --sc-rebalance-delay
isi network pools modify groupnet0.subnet1.pool0 –sc-auto-unsuspend-delay
```

3.2.6.4 Update the tunable

```
isi network external modify --sc-rebalance-delay=1
```

```
isi network pools modify groupnet0.subnet1.pool0 –sc-auto-unsuspend-delay=2
```

3.2.7 Asynchronous delete

This setting was introduced in OneFS version 8.2, and it is used with small-file EDA workloads to speed up the delete operations. It performs asynchronous deletes to significantly improve the latencies involved in delete operations.

3.2.7.1 Effects on OneFS

This setting would allow faster deletes for many EDA workloads resulting in improved user job performance.

3.2.7.2 Default settings

By default it is set to 1 which means it is disabled. Use the below command to check the current setting:

```
sysctl efs.bam.async_delete_mode
```

3.2.7.3 Recommended settings

For EDA environments where performance of small file deletes are of importance, then the setting the async_delete_mode to 2 is recommended.efs.bam.async_delete_mode

3.2.7.4 Update the tunable

```
sysctl efs.bam.async_delete_mode=2
```

3.3 Expert-level settings

Do **not** apply the settings in this section without the guidance of an SE and a technical support personnel. Please open a Service Request (SR) prior to making any changes.

3.3.1 LWIO

OneFS uses a form of LWIO settings to schedule I/O to NFS, SMB, and HDFS. When a client connects, it registers with a front-end LWIO container. As the client issues I/O, the LWIO container tracks it.

Response time is measured from when the request is received to when we it is satisfied, including any queuing on the front-end. Queue time and service time are not individually measured but are included in the overall response time.

3.3.1.1 Effects on OneFS

In storage systems, it is common to restrict the number of parallel sessions reading or writing at the same time. While having more parallel streams is generally good, too many streams can hurt performance. The number of threads allowed is limited in the getblk or get block state on a per-node basis. This is limited by a multiplier based on the number of cores a node has.

In some situations with certain I/O profiles, the default number of threads do not allow enough parallelism. This results in high response times for the system, while it appears to have low CPU and disk usage (average time in queue and average pending I/O operations).

LWIO settings can be increased with SE or CAE guidance.

3.3.1.2 Default settings

```
registry.Services.lwio.Parameters.Drivers.onefs.OnefsCpuIoMultiplier (uint32) =
2
registry.Services.lwio.Parameters.Drivers.onefs_functional.OnefsCpuIoMultiplier
(uint32) = 2
registry.Services.lwio.Parameters.Drivers.onefs_hdfs.OnefsCpuIoMultiplier
(uint32) = 3
registry.Services.lwio.Parameters.Drivers.onefs_lwswift.OnefsCpuIoMultiplier
(uint32) = 3
registry.Services.lwio.Parameters.Drivers.onefs_nfs.OnefsCpuIoMultiplier
(uint32) = 3
```

There are few settings available with this tunable. If changing the multiplier for an A200 node, it should be changed for both NFS and SMB. These are set to 3 and 2 parallel I/Os, respectively. This number is the number of I/Os. For A200 nodes using NFS and SMB, the starting multiplier should be 4 for both (3 to 4, and 2 to 4). This is applicable to A2000 nodes as well. This creates some equivalency with an NL410 system. Increasing this multiplier results in busier disks. If disks are already 50–60% utilized, stop increasing the multiplier. Otherwise, the setting can be adjusted above 4. The other multipliers for HDFS and swift could also be increased to 4 at first, if those protocols are in use.

The standard settings are as follows:

```
isi_gconfig | grep -i onefscpu

registry.Services.lwio.Parameters.Drivers.onefs.OnefsCpuIoMultiplier (uint32) =
2

registry.Services.lwio.Parameters.Drivers.onefs.OnefsCpuMultiplier (uint32) = 4
```

3.3.1.3 Recommended settings

This setting is based on trial and error. Using too high of a setting causes the system to become overutilized. This issue has been noticed on A200 nodes with CommVault workloads, since work typically traverses any available thread that CommVault has available to it.

This type of I/O does not keep the standard A200 configuration busy; the node simply has low CPU and low disk usage (average pending operations, average time in queue, average service time).

Increasing the count 2x or even 4x can significantly help performance of an A200 node for CommVault workloads.

Note: This tunable is primarily focused at the lower performance nodes. In some situations, other node types with certain workloads may benefit

---

Check the current settings:

```
isi_gconfig | grep -i onefscpu

registry.Services.lwio.Parameters.Drivers.onefs.OnefsCpuIoMultiplier (uint32) =
2

registry.Services.lwio.Parameters.Drivers.onefs.OnefsCpuMultiplier (uint32) = 4
```

Based on this number above (2), an A200 node with a 2-core CPU has 4 threads in getblk.

Show the number of sessions blocking per node:

```
isi_for_array 'pgrep -f "lw-container lwio" | xargs procstat -t | grep getblk |
wc -l'
```

Show the total number of sessions on each node:

```
isi_for_array 'pgrep -f "lw-container lwio" | xargs procstat -t | grep lwio | wc
-l’
```

If the number is constantly at 4 or above, and the response time is high with the other stats being low, LWIO starvation looks likely.

3.3.1.4 Update the tunable

Increase the thread limits to the first level. Instead of disabling or enabling SMB or NFS, perform an LWIO restart using **/usr/likewise/bin/lwsm restart lwio.** This only performs the restart on this node, and allows it to be performed on each node manually or with **isi_for_array**.

```
isi_gconfig
registry.Services.lwio.Parameters.Drivers.onefs.OnefsCpuIoMultiplier=4

isi_gconfig registry.Services.lwio.Parameters.Drivers.onefs.OnefsCpuMultiplier=8
```

Disable SMB or NFS (SMB shown):

```
V8111-1# isi services smb disable

The service 'smb' has been disabled.

Wait a minute and then enable it again

V8111-1# isi services smb enable

The service 'smb' has been enabled.
```

Show the number of sessions blocking per node (should be higher):

```
isi_for_array 'pgrep -f "lw-container lwio" | xargs procstat -t | grep getblk |
wc -l.
```

After increasing the LWIO settings, assuming the application is able to drive more workload, there is an increase in CPU and disk activity. The system appears to become more utilized. Make sure this usage is not excessive and allows headroom for system jobs or node failures.

3.3.2 F/H and A series nodes in a cluster

It is common for Isilon solutions to use tiering that combines a faster tier and an archive tier. An extreme example of this would be an F800 with an A2000 node.

A common question regards using GNA for an A2000 node to increase metadata performance. This is not possible because the A2000 node has SSDs used strictly for L3 cache, which only targets caching metadata for this node type. While this has a high chance of caching metadata entries in a low-change-rate workload,

there is another way to potentially boost performance, especially for tasks that involve tree walks. This method involves placing directory metadata on the higher tier and file data on the lower tier.

3.3.2.1 Effects on OneFS

When a user process performs a task like a tree walk (for example, a find operation), directory metadata is scanned to show the various files in directories. While some processes individually stat files, many do not and rely on directory metadata performance.

When using a filepool policy to tier data, target only files and leave directories (the metadata for them) on the faster tier. This can speed up metadata operations.

3.3.2.2 Recommended settings

A node such as the A2000 should have a low change workload tiered to it, especially if the access time is set modestly. This allows access to a file without incurring a metadata write, in which the metadata entry remains current in either L2 or L3 cache. For the best performance, consider targeting type = files in the file policy pool to leave the directory metadata on the higher performing node pool.

Use the filepool policies in the WebUI to check this tunable for it's current values.

3.3.2.3 Update the tunable

Use the standard option in the WebUI to check this tunable.

This setting is best verified end-to-end with a process such as a find command over NFS. This shows that the directory structure enumerates faster. Certain operations like recursive listings perform stats, and these rely on metadata from the archive tier.

3.3.3 Additional NFS settings

There are three settings related to client performance that are presented here as recommendations for EDA.

3.3.3.1 Effects on OneFS

These settings affect the client performance.

3.3.3.2 Default settings

To view the default settings for each setting run following commands:

```
isi_gconfig registry.Services.lwio.Parameters.Drivers.nfs.nlm.QuarantineTimeSec
isi_gconfig registry.Services.lwio.Parameters.Drivers.nfs.nlm.RpcIdleTimeout
isi_gconfig
registry.Services.lwio.Parameters.Drivers.nfs.SecurityCacheUidGidLruSize
```

3.3.3.3 Recommended settings

The values recommended are 30, 360, and 4000 respectively for QuarantineTimeSec, RpcIdleTimeout, and SecurityCacheUidGidLruSize settings above. Please note, the recommended setting of 30 seconds for the QuarantineTimeSec setting may change in future.

3.3.3.4 Check the tunable

```
isi_gconfig registry.Services.lwio.Parameters.Drivers.nfs.nlm.QuarantineTimeSec
isi_gconfig registry.Services.lwio.Parameters.Drivers.nfs.nlm.RpcIdleTimeout
```

```
isi_gconfig
registry.Services.lwio.Parameters.Drivers.nfs.SecurityCacheUidGidLruSize
```

3.3.3.5 Update the tunable

```
isi_gconfig
registry.Services.lwio.Parameters.Drivers.nfs.nlm.QuarantineTimeSec=30isi_gconfi
g registry.Services.lwio.Parameters.Drivers.nfs.nlm.RpcIdleTimeout=360
isi_gconfig
registry.Services.lwio.Parameters.Drivers.nfs.SecurityCacheUidGidLruSize=4000
```

4 Client access

This section introduces best practices for setting up client access.

4.1.1 NFS considerations

NFSv3 is the ubiquitous protocol for clients accessing storage. This is due to the maturity of the protocol version, ease of implementation, and wide availability of client and server stacks.

There are some important EDA configuration settings to keep in mind when using Isilon with NFS clients in an EDA environment, which are detailed in the following subsections.

4.1.2 Client NFS mount settings

For NFS3 and NFS4, the maximum read and write sizes (rsize and wsize) are 1 MB. When mounting NFS exports from a cluster, a larger read and write size for remote procedure calls can improve throughput. The default read size in OneFS is 128 KB. An NFS client uses the largest supported size by default. Setting the value too small on a client overrides the default value and can undermine performance.

For EDA workloads, the recommendation is to avoid explicitly setting NFS rsize or wsize parameters on NFS clients when mounting Isilon NFS exports directly, or using the automounter. Instead, for NFSv3 clients, use the following mount parameters, either directly or in the NIS Automounter map:

```
mount -vers=3,rw,tcp,hard,intr,retry=2,retrans=5,timeo=600
```

With NFS clients that support READDIRPLUS, this call can improve performance by prefetching the file handle, attribute information, and directory entries, plus information to allow the client to request additional directory entries in a subsequent readdirplus transaction. This relieves the client from having to query the server for that information separately for each entry.

For an environment with a high file count, try setting the readdirplus prefetch to a value higher than the default value of 10. For a low-file count environment, experiment with setting it lower than the default. In a workload that runs concurrent jobs, consider testing the changes until identifying the value that works best for the environment.

Find more information about readdirplus in the Dell EMC KB article emc14001899, “Directory listing is slow with a large amount of files using Linux clients”.

Another recommendation for EDA is to use asynchronous (async) mounts from the client. Conversely, using sync as a client mount option makes all write operations synchronous, usually resulting in poor write performance. Sync mounts should be used only when a client program relies on synchronous writes without specifying them.

4.1.3 NFS connection count

As a conservative best practice, active NFS connections should be kept under 1,000, where possible. Although no maximum limit for NFS connections is established, the number of available TCP sockets can limit the number of NFS connections. The number of connections that a node can process depends on the ratio of active-to-idle connections as well as the resources available to process the sessions. Monitoring the number of NFS connections to each node helps prevent overloading a node with connections.

4.1.4 NFSv3 or NFSv4

NFSv4 can provide some benefits over NFSv3 due to a slight performance advantage when working with large clustered environments accessing a common resource.

NFSv4 provides several new features as well as improvements on the NFSv3 architecture. When working with a large distributed data management infrastructure, NFSv4 provides the following major advantages over v3:

- A more stateful implementation
- Ability to bundle metadata operations
- An integrated, more functional lock manager
- Conditional file delegation

NFSv4 now requires that all network traffic management (congestion, retransmits, timeouts) be handled by the underlying transport protocol as opposed to the application layer as found in v3. In high-volume, high-throughput workflows, this helps free up the client for additional application specific work on the data.

NFSv4 also has the ability to bundle metadata operations using compound remote procedure calls (RPCs), which reduce the overall number of metadata operations and significantly decrease the overhead required when accessing multiple files. This can be a significant factor for data-management frameworks, which often require access to hundreds or even thousands of files to satisfy client requests.

An integrated lock manager provides lock leasing and lock timeouts, a considerable improvement over the previously used NLM in NFSv3, which only provided a limited implementation of these features out of band. This enables cleaner recovery semantics and processes for failure handling.

File delegation is another new feature in NFSv4 in which the server provides a conditional, exclusive lock to the client for file operations.

4.2 Permissions, auth, and access control

4.2.1 NIS and access zones best practices

A minimum of two NIS servers provides redundancy and helps avoid access-control lookups from being a bottleneck. For larger environments, scaling the number of NIS servers may be required.

The maximum number of supported NIS domains is 50.

Although multiple NIS domains can be specified in an access zone, NFS users benefit only from the NIS configuration defined in the system access zone.

As a best practice, the number of access zones should not exceed 50. The number of local users and groups per cluster should not exceed 25,000 for each. While possible, creating a larger number of local groups or users may affect system performance.

4.2.2 Group owner inheritance

Many EDA implementations use group IDs to protect project data. To configure group IDs to work well with EDA environments, the following ACL configuration change on the cluster is recommended.

To configure from the WebUI, navigate to Access > ACL Policy Settings > Group Owner Inheritance, and select Linux and Windows semantics > Inherit group owner from the creator's primary group.

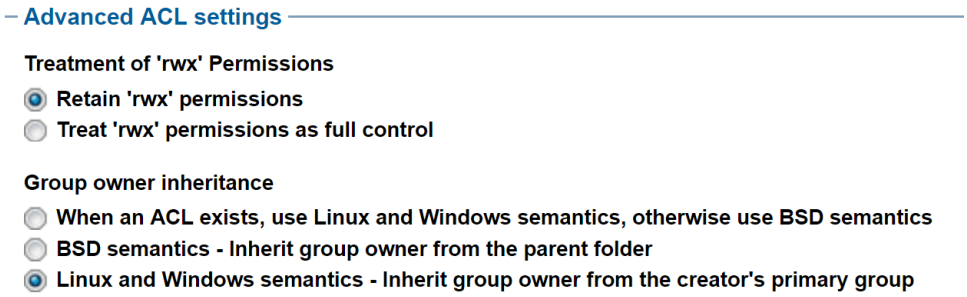

Figure 4: Group Owner Inheritance Configuration

This allows for the child entities created to inherit the parent attribute of effective-GID, rather than always inheriting the primary-GID.

5 Data Management

This section discusses data management best practices including tiering, managing data by node type, SmartPools, archiving, and other topics.

For more detailed information on SmartPools and EDA, refer to the white papers and best practices documents listed in the references section. These documents delve into general best practices or practical implementation aspects of the Isilon scale-out storage system in semiconductor chip design environments. For storage administrators, it is recommended to refer to additional documentation to cover other areas of Isilon storage such as SmartConnect, SnapshotIQ, or SyncIQ to gain more comprehensive knowledge. Adhering to best practices in just one functional area, such as SmartPools, in isolation may not yield the desired and optimal cluster functionality.

5.1 Tiering

Having the right type of data on the right type of storage sub-system at the right time is tiering. Tiering exists to achieve the business objectives out of storage system with minimum overhead. The primary business objective usually is to avoid or reduce storage expenditure for non-business-critical data, driving the Total Cost of Ownership (TCO) lower. Another somewhat subtle driver for tiering is that semi-conductor IC designers are often more efficient spending their time on design work instead of cleaning up data. Due to the design data being inter-woven in complex directory structures and very large number of files, it could be time consuming to keep inactive data out of performance tier. Hence, as the accumulated data ages, it becomes increasingly difficult to assign clean-up responsibilities to designers or for IT to identify removable data. On top of that staff attrition makes it almost impossible to delete data over time. Yet another driver for tiering is tools and their versions. Chips are designed using specific tools, versions and associated libraries, so designers tend to save entire design projects to avoid tool mismatch or unavailability in future. Lastly, if a design must be re-spun in future, it is far easier if entire project was saved rather than just the source files.

While automated policy-based tiering helps in driving the TCO down, chip designers would like to improve their job runtimes regardless of location of data inside the cluster. Hence, depending on the amount of active Vs semi-active Vs inactive data, business archive policies, performance requirements, TCO requirements, and so on, effective tiering requires a careful consideration of how to mix node types in the cluster.

5.2 Managing Data by Node Types

Isilon provides unmatched scale and versatility in meeting business goals. With nodes ranging from all-flash to hybrid to archive, there can be many ways the storage community can design and operate the Isilon clusters for their businesses. However, in semiconductor CAD workflows there are enough similarities in operating environments across the spectrum of companies that some specific configurations can be called out as best practices.

Given below are three node-pool combinations that design houses could consider based on the suitability of their workloads. Please note, these options are based on the current Isilon node types available, and these are subject to change. We will discuss in brief some of the factors that influence the selection of node types.

5.2.1 Option 1: All-flash cluster

Some chip-design or chip manufacture companies have embarked on the all-flash data center initiatives. The TCO is viewed in a different light here; essentially through improving EDA job run times, which has a downstream effect of reducing time to market for the designs as well as reducing the burden of expensive EDA tools licensing costs. In addition, reducing physical data center space, power, and cooling, are added

tangible benefits. Moreover, subtle benefits such as faster turnaround time for chip defect management, simplicity of managing consolidated, homogeneous and reduced footprint are significantly augmented.

However, one cannot escape the fact that in semiconductor, roughly 70% of all data is not in active use, and data losing its business-critical value is a perpetual event. Consider this, if a 5PB storage footprint grows at 30% every year, every month 125TBs of new ingest space is expected. Knowing that data becomes stale almost continuously, if there's no Information Lifecyle Management strategy in place, there would be no egress of data from primary storage tier, resulting in a 30% growth of all-flash environment. This could quickly catch-up with the business management. Hence for an all-flash strategy, it's crucial to have a solid ILM, or a basic archive strategy in place. More on archives in a separate section.

In all, for an all-flash cluster (i.e., all node types are F800 or F810), there are no user-defined SmartPools file pool policies involved, except the Default Policy. It should be setup as follows:

- In the default file pool policy, ensure the Set Data Access Pattern is set to 'Optimize for concurrent access' – this invokes adaptive prefetch algorithm

Use SyncIQ to move archive data out of the primary cluster on a regular basis. Alternatively, use ClarityNow to copy or move data based on user-defined criteria.

The all-flash cluster could also be created with the Isilon F810 nodes which provide hardware-based inline compression. A homogeneous F810 based cluster is beneficial for semiconductor due to the overall compressibility of all EDA sub-data sets. Compression can be leveraged to realize additional storage density benefits beyond what the SSD capacities offer. More importantly, it promotes higher consolidation of the storage hardware footprint, along with savings in power, space, etc. This consolidation further negates the need to mix archive or high-density nodes with the all-flash nodes. With growing semi-conductor design densities, chip design work requires more storage, and having a single namespace hosting large partitions is very valuable to designers.

There are two things to keep in mind with F810 all-flash cluster. Ensure that the workflows you plan to host on this cluster have compressible data. Based on experience and some internal testing, semiconductor CAD workflows are generally found to be compressible but having some estimate would help you size the cluster right for current and future needs. Second, the SSD capacities range from about 3TB to 15.36TB per drive. From a IOPs/TB perspective, the 7.68TB SSD disk would be recommended as a sweet spot for primary workflows.

Also note, currently, maximum of 40 F810 nodes can be supported (or 10 chassis) within a single cluster.

5.2.2 Option 2: Hybrid cluster

In this scenario, the cluster is composed of some F800 nodes addressing the active dataset, and the Isilon hybrid nodes (H500) addressing the rest of the datasets (try not to include archive in this). This option requires less capital outlay and with correct placement of data, reduced compute job runtimes. All active data should fit into the F800 node pool with some space left for growth. Consider this example:

Total usable space needed = 1000TBs, with about 300TB expected to be active or business critical

Using the 3.84TB disk option, this would require about 6 F800 nodes and using the 4TB HDD option, would require 12 H500 nodes, for a total of 18 nodes in the cluster. This achieves approximately a 30:70 ratio of flash Vs hybrid space. All the file pool policies you would design for the placement of data in the two tiers should keep this ratio in consideration. If your workloads require a 40:60 ratio or 50:50 ratio that is fine, but more than 50% flash could be questionable and tends to favor the all-flash option.

Here are some things to keep in mind –

- If this 30:70 ratio (or any) is to be maintained, an archive policy must be in place, without which the hybrid nodes will accumulate more data with time resulting in ratios of 20:80 or 15:85 ratio, which is deviation from the original design goal. In absence of data archival, if you try to maintain the 30:70 ratio forcefully by adding more F800 nodes, whenever H500 nodes are added, that could end with an under-utilized flash tier. Overall, if you add nodes on either tier on an as-needed basis without regard to maintaining the ratio, or without an archive policy, the resulting deviation from original design goals would require tweaking file pool policies more often.
- This option is really hosting active and semi-active data, not active/inactive. Carefully decide what falls in each category. For e.g., make a table of your data categories and assign them a business criticality from the standpoint of performance, and data protection:

Project directories (/project/projabc)	**Fastest, snapshots and backups**	**Tier based on atime or mtime down to H500**
Scratch areas (/scratch/projabc_scratch1)	Fastest (?), no snapshots (?), no backups	Pin to F800 or H500 by path
EDA Tools and Binaries (/tools/tools1)	Medium, less frequent snaps and backups	Pin to H500 by path
Foundational IP (/project/iplib1)	Medium, no snaps, less frequent backups	Pin to H500 by path
Home directories (/home/homedir1)	Fastest (?), snapshots and backups	Pin to F800 by path
Smb-only directory (\\cifs.cluster.com)	Medium, snapshots and backups	Pin to H500 by path
Others	Performance? snapshots? Backups?	As needed

Table 1: Categorizing Data, Determining its Placement and Data Protection Strategy

- As data ages from projects or as older tools and libraries are retired they become archive candidate. Use the Isilon SyncIQ feature to archive them to another high-capacity cluster (say all A2000 nodes). Maintain this egress of data continually to keep up with the growth rate. In addition to SyncIQ, Dell EMC ClarityNow is a very useful software application that can be leveraged to achieve some very meaningful archive policies.
- Also note, since scratch may not be a good archive candidate (if truly used as a scratch space), it can be auto-tiered, and users could be persuaded to delete data periodically. Perhaps this is the only data in semiconductor CAD design work flows that could be deleted periodically.

5.3 SmartPools Best Practices

Consider the below configurations whenever there are two node types involved in the manner described above. Keep in mind, conceptually, you want the F800 nodes to act as your landing zone for ingest of data:

1. Change the Default Policy
    a. Set Storage Target to the F800

b. Set Snapshots Storage Target to F800 (Note: There is a trade-off here between space and performance. Storing snapshots on a high-capacity tier could incur the snapshot Copy On Write performance penalty. The gain would be in saving some space on flash tier)
c. Change the SSD strategy to 'Use SSDs for Data and Metadata' – this is needed to use the F800s for active datasets in this scenario
d. Set Data Access Pattern to 'Optimize for concurrent access' – this invokes adaptive prefetch
e. "Enable SmartCache" should be selected

2. Create user-defined policies pretty much in the order shown in the table 1 above

a. For each policy, define the path-based tiering. For e.g., "if path matches /ifs/cluster/edatools then Storage target is H500 pool, SSD strategy is 'Use SSDs for data and metadata,' Snapshot storage target is H500 and its SSD strategy is 'SSD strategy is 'Use SSDs for data and metadata' and the data access pattern set to concurrent access as shown below in figures 5 and 6:

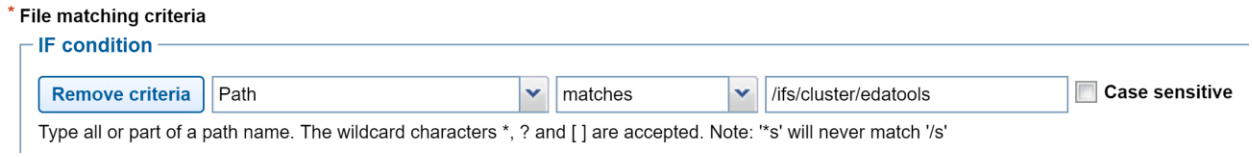

Figure 5: User-defined FilePool Policy Path Settings

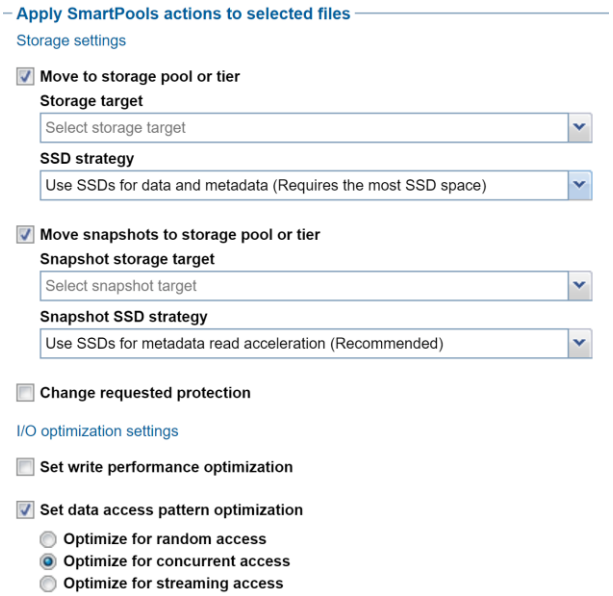

Figure 6: User-defined FilePool Policy Settings

- Semiconductor CAD workflow datasets are extremely meta-data oriented. Under the SmartPools/Node Pools settings, disable the L3 caching option on both node pools
- Under SmartPools settings tab, uncheck the "Use SSDs as L3 cache by default for new node pools"
- If both node types were purchased together and you're creating a new cluster, make sure that the F800 nodes are installed first, and then add all other nodes in including H500 to the cluster. This is to keep the 'system' disk pools on the F800s. As a result, the /ifs/.ifsvar will be placed on F800s which is preferred

5.4 Other considerations

There are generic SmartPools best practices discussed in detail in the SmartPools white paper available at https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h8321-wp-smartpools-storage-tiering.pdf. Below list summarizes some of them for quick reference:

- Ensure that cluster capacity utilization (HDD and SSD) remains below 85% on each pool
- If the cluster consists of more than one node type, direct the default file pool policy to write to the higher performing node pool. Data can then be classified and down-tiered as necessary
- A file pool policy can have three 'OR' disjunctions and each term joined by an 'OR' can contain at most five 'AND's
- Define a performance and protection profile for each tier and configure it accordingly
- File pool policy order precedence matters, as the policies are applied on first match basis (i.e., the first file pool policy to match the expression will be the applied policy).
- If you create a file pool policy to be run at a higher frequency, ensure the SmartPools job is configured to run multiple times per day.
- This ensures that there's space available for data reconstruction and re-protection in the event of a drive or node failure, and generally helps guard against file system full issues.
- Avoid creating hard links to files which will cause the file to match different file pool policies
- If node pools are combined into tiers, the file pool rules should target the tiers rather than specific node pools within the tiers.
- Avoid creating tiers that combine node pools both with and without SSDs. The number of SmartPools tiers should not exceed 5.
- Where possible, ensure that all nodes in a cluster have at least one SSD, including nearline and high-density nodes.
- For performance workloads, SSD metadata read-write acceleration is recommended. The metadata read acceleration helps with getattr, access, and lookup operations while the write acceleration helps reduce latencies on create, delete, setattr, mkdir operations. Ensure that sufficient SSD capacity (6-10%) is available before turning on metadata-write acceleration.
- Avoid using the 'isi set' command or the OneFS Filesystem Explorer to change file attributes, such as protection level, for a group of data. Instead use SmartPools file pool policies.

5.4.1 Option 3: Homogeneous Cluster

Like the all-flash option, some customers have chosen to leverage the cost-capacity-performance equation of the H500 node for their workloads. The advantages to this approach is of course lower capital outlay, and higher capacity, provided there is a good alignment of performance expectations with the node's CPU and memory specifications. We see customers deriving much success with this type of cluster, however, with this approach, it requires careful planning on workload consolidation and a good understanding of the workflows of the environment. Also, as with any platform, exercise good judgement when going with the higher density drives, as IOPs/TB tends to go lower with denser disk drives.

5.5 Other considerations

- Semiconductor CAD design workflows are unique because the storage foot print is a combination of active, semi-active (including read-only), and cold data (archive candidates). The three options presented here are best suited for this type of data distribution. While other verticals such as media and entertainment see the combination of all-flash and archive nodes (F800 with A200 or A2000) as a valuable configuration, this is not recommended.

- As of the writing this document, hardware-based compression is only supported for the F810 nodes. When these nodes are mixed with other nodes such as the H500 or even the F800, some additional CPU cycles are spent on compression/decompression of data as it traverses out of or into the F810 nodes as part of a file pool policy. The performance impact could vary based on workloads and node types and could be even negligible, but for the performance savvy semiconductor customers that is vying the data reduction benefits with all-flash level performance may see mixing other non-flash nodes as a contradictory strategy. For e.g., the F810 node with 7.63TB SSD with a 2:1 compression ratio yields 15TB+ effective capacity per drive, which is far more than what an H500 offers with spinning disks, just from a capacity stand point.
- Ensure that at least 2 SSDs are configured per node (for the H500) or the larger capacities. Refer to the resources section for further information on sizing the cache SSDs or work with your SE for appropriate sizing. For semiconductor, experience indicates that picking two largest SSDs available could allow the option of using the “data and metadata” SSD strategy in the file pool policies
- There is a new H5600 node available (Ethernet back-end only). This node combines the performance of an H600 node with the storage capacity of an A2000 node. The storage capacity offered is based on the 10TB SATA drives in a deep chassis. Such a performance-capacity combination is ideal for workloads in Advanced Driver Assistance Systems (ADAS) development, active archives, media and entertainment, etc. This node can be attractive option in semiconductor for hosting semi-active data sets that are not sensitive to IOPs/TB ratios. With careful planning around client (compute) operations, and nature of workloads, this new node could potentially be an alternative to H500 node in options 2 and 3 discussed above. In addition, it would be an ideal choice for active-archive clusters, as well as for backup clusters

5.6 Archives in Semiconductor

In semiconductor, during chip design, the business-critical data related to that design is stored in various project partitions. One design project could be spread out over many partitions ranging from a few GBs to 100TB or more. These partitions that constitute a “whole” design may contain a mixture of lot of small files and some large files. In some cases, a design environment has a best practice of storing all necessary design tools along with the project data, and in other cases, tools are maintained separately and centrally. The point is, a full design spans one or more partitions, is large, and contains many millions of files.

While Isilon’s single namespace significantly reduces the need for many smaller partitions, some partitioning is necessitated based on design methodologies adopted by each company. Overall, due to the wholesome nature of hosting designs across partitions, it makes sense to archive the design work by saving all the partitions as-is, so that few years down the road if a design needs a re-spin, the data layout (including any soft/hard links) is intact in archives, with the directory hierarchies preserved just the way it was originally. This provides most flexibility to the designers and shortest time to resume work on the design once it is unarchived. In contrast, when archive policies are designed around access time of files, it may end up archiving parts of a project, leaving behind some files in the original production partition.

Unix is prevalent in semiconductor environments, and the usage of autofs (automatic mounting of network filesystems) functionality integrated with the NFS export rules and authentication providers seems to be a norm. With thousands of partitions for projects, scratch spaces, tools, homes, it can be daunting for the Unix admin to maintain a clean set of automounter maps.

With the above information around design archives and associated Unix environment challenges, following best practices for semiconductor archives can be considered.

- Provision a separate cluster for archives using capacity nodes such as the A2000, or A200. Note, a cluster that is designed to host SyncIQ backups could also be used for archives

  a. If archives are to be protected, SyncIQ can be used to protect it on another cluster or to an Isilon cluster in public cloud
  b. Alternatively, the Dell EMC's ECS object storage platform can also be an archive target, providing geo replication and global namespace capabilities among other benefits
  c. The goal is to provide online archive capabilities, which is an incentive to the designer to archive more, knowing the data is available instantly to read

- Create a separate map such as /archives for all projects. For e.g., it can be available to designers as /archives/projectabc for a project, /archives/projectabc_scratch for the associated scratch partition, and /archives/projectabc_tools for specific tools that could be archived
- Have a separate data lifecycle policy and process for defunct home accounts, and retired tools and libraries
- The data under /archive should be available as read-only, while the top-level archive directory on the archive cluster is available as read-write export for administrative hosts

Data classification is key to successful archive policies in the company. Giving the designers ability to see the aging of their data, being able to manage their own data classification, tagging their partitions, and letting them archive it on a routine basis are all good measures for an effective data life cycle management. Furthermore, the data movement for archive or workflow collaboration can be automated and integrated within the IT processes. Dell EMC offers a software application called ClarityNow. It can easily scale to handle billions of objects and present the data about data to IT and end-users to achieve effective life cycle management in semiconductor.

5.7 Alternate Ways to Tier Data

Intelligent tiering of data is applicable to the mixed-node cluster option discussed earlier. The SmartPools feature is utilized to achieve the business objectives. The SmartPools job executes the file pool policies including the default file pool policy. These policies define the data protection settings, caching strategies, data ingest, snapshot data placement, and so on. The reader is advised to review the SmartPools Best Practices document which describes in detail many aspects of this important feature.

One thing to note is that the scope of this job is all the nodes of a cluster. In other words, the job executes a LIN tree scan across all the nodes in parallel by breaking up the work across the nodes. When a policy attribute match is found against a file, it stops processing the policy for that file as that is the first policy that hits a match. The job then checks the current settings of the file and compares with desired settings stated in the policy and applies them. It would also restripe the data to reflect any changes in protection scheme, etc. On a fairly large mixed-node cluster with a 70-30 ratio of hybrid-flash nodes, SmartPools could take longer to finish especially if the file count is very high.

In semiconductor, the primary use case for SmartPools daily is to execute tiering from performance to capacity node pool (or tier) or vice versa. If the file pool policies that you have defined are not changing the protection settings for a given path, then the SmartPoolsTree job can be run instead of SmartPools. This new job was introduced in the 8.0 code family to selectively run the file pool policies. Here's an example:

```
# isi job jobs start SmartPoolsTree --policy=HIGH --paths=/ifs/acd8007/projects
```

So, for example, if there is a file pool policy already defined with the above path that takes a certain action (such as setting data access pattern to “random”), then the above SmartPoolsTree job would achieve the same end-result as SmartPools job as defined by the file pool policy. See the corresponding policy below:

```
# isi filepool policies view test
                                Name: test
                         Description: -
                               State: OK
                        State Details:
                         Apply Order: 2
                File Matching Pattern: Path == acd8007/projects (begins with)
             Set Requested Protection: -
…
```

Note that if there is no file pool policy available with the path “8007/projects” then the SmartPoolsTree job would go by the default file pool policy and apply the settings from that policy. In above example, if the policy “test” which sets Data Access Pattern to “random” were NOT available, then the SmartPoolsTree job would end up setting Data Access Pattern and other settings as dictated by the default policy, which in this case is “concurrent.”

The SmartPoolsTree job can also be used to quickly set ingest policy on directories, execute a dry run, apply to directories only, or apply a policy recursively. This job can be scheduled to run at specific times or can be run manually. For semiconductor CAD environment, if a cluster is hosting billions of small files, then depending on the node types in use, SmartPoolsTree jobs can be created if the SmartPools job is found to be taking too long to complete. Do not delete the file pool policies, rather, the schedule of SmartPools job can be changed so that it runs less frequently.

Note: A SmartPools license is needed to use this job.

In OneFS 8.2, a new job is introduced, called FilePolicy. This job delivers a more streamlined, lower cost (performance wise) way to apply filepool policies. It has improved scan performance and delivers tiering at reduced impact on CPU and disk. This job does not replace SmartPools job, which is still needed to apply protection settings or other criteria outside of tiering. However, since it’s focused on tiering and is not tied to full cluster LIN scans, it can tier faster than SmartPools. Details of this new job will be released post OneFS 8.2 release as part of updated existing documents or new document.

Data Protection

Protecting semiconductor design or manufacturing data is no less important than in any other industry. However, given the wide variety of workloads and design methodologies in semiconductor ranging from project data to home directories, tools repositories, and data related to production of silicon, different methods can be applied to protect the data. While Dell Technologies offers a variety of products and solutions to protect the data, this document is focused on the Dell EMC Isilon based solutions.

5.7.1 Business Service Level Agreements (SLAs)

It’s important to first understand which datasets the business (i.e., management) considers as worth protecting and for how long. Unlike finance industry, data protection and retention requirements in semiconductor are not governed by regulatory agencies. Often, this is driven by either customer contracts, or basic industry standard protection is assumed. Hence, some customers retain completed designs for 7, 10 or 15 years after final tape-out or a certain period after first production batch. Given the semiconductor storage

requirements increasing exponentially with higher design densities, knowing what not to protect is critical so that data that does need protection can be adequately protected. For example, depending on the design practices employed, there may not be any need to protect the so called 'scratch' data. This could significantly change both architecture and budget requirements. Once you know all datasets that must be protected, Service Level Agreements (SLAs) can be reviewed to decide on retention periods, restoration time requirements, encryption, and so on. As such, the SLA itself can indicate what the business must protect.

5.7.2 What to protect?

The table below shows one way to organize the chip design data. It lends itself well in terms of data protection and management:

Path (example)	Protection Types	Sample Data Protection Scheme
Project directories (/project/projabc)	Business-critical design data	Short-term, and long-term
Scratch areas (/scratch/projabc_scratch1)	Critical but reproducible, transient data	No protection needed or only short-term
EDA Tools and Binaries (/tools/tools1)	Mostly read-only data	Weekly backup, no snapshots
Foundational IP libraries (/project/iplib1)	Mostly read-only data	Weekly backup, no snapshots
Home directories (/home/homedir1)	Business-critical data	Daily protection, onsite and offsite
Smb-only data (\\cifs.cluster.com)	Business-critical data	Daily protection, onsite and offsite
Others (repositories, software development)	Business-critical data	Daily protection, onsite and offsite

Table 1: Organizing Chip Design Data

Scratch data is a big constituency among all data, normally containing data that is transient in nature. If It can be used as a dumping ground for large log files generated during various design phases. In terms of storage allocation and accounting, this space is usually allocated more generously by IT and cheaper compared to a project partition in lieu of no extra cost for protecting the data. Hence, IT expects that for some reason this data is lost, it should not cause a delay in the design cycle. Therefore, the end-users must be well aware of the IT best practices and data protection policies offered. Optionally, short-term snapshots can be configured for scratch areas.

5.8 Protection Methods

Approaches to data protection start from fault tolerance on the storage systems, to snapshots, replication (local and/or geographically separate), and backups to nearline storage, VTL, or tape. Some of these methods are biased towards cost efficiency but have a higher risk associated with them, and others represent a higher cost but also offer an increased level of protection. Once again, the IT management would need to first define what are the defined and documented SLAs and where do they apply. Two ways to measure cost versus risk from a data protection point of view are:

- Recovery Time Objective (RTO): RTO is the allotted amount of time within a Service Level Agreement (SLA) to recover data. For example, an RTO of four hours means data must be restored and made available within four hours of an outage.
- Recovery Point Objective (RPO): RPO is the acceptable amount of data loss that can be tolerated per an SLA. With an RPO of 30- minutes, this is the maximum amount of time that can elapse since the last backup or snapshot was taken.

Based on the table above, short-term retention, which is usually based on snapshots is used for specific data sets to satisfy low recovery objective SLAs. For projects, and home directories, replication of data from the primary cluster to a target DR cluster, ideally located at a geographically separate location, is strongly recommended. NDMP backup to tape or VTL (virtual tape library) typically satisfies longer term high recovery objective, disaster recovery SLAs and any regulatory compliance requirements.

Below is a sample recommendation for different data types in semiconductor:

Project areas: Snapshots, replication, backups, and off-site storage

Scratch areas: No protection beyond system fault tolerance configuration

EDA tools, binaries: Snapshots, replication (for distribution), less frequent backups

Libraries: Replication (for distribution), less frequent backups

Home directories and project accounts: Snapshots, backups, and off-site storage

Other data: define policies based on business SLAs

Dell EMC Isilon offers the SnapshotIQ feature for snapshots, and SyncIQ for replication and backups (to another Isilon). The Isilon OneFS supports NDMP protocol for both a 2-way and 3-way backup configuration. The Dell EMC suite of products include backup solutions that can be engaged in the overall Isilon data protection architecture. In addition, there are a number of 3rd-party backup applications that can utilize the SnapshotIQ feature to backup data on to Dell EMC ECS Object Storage platform, Virtual Tape Libraries, physical tape devices or to cloud. Each solution addresses specific business objectives for the protection of data.

5.8.1 Backups and replications with SyncIQ

Isilon SyncIQ delivers high-performance, asynchronous replication of unstructured data to address a broad range of recovery point objectives (RPO) and recovery time objectives (RTO). This enables customers to make an optimal tradeoff between infrastructure cost and potential for data loss if a disaster occurs. SyncIQ does not impose a hard limit on the size of a replicated file system so will scale linearly with an organization's data growth up into the multiple petabyte ranges.

SyncIQ can be leveraged to protect data in several different ways starting from replicating EDA Tools and binaries to sharing project areas across different geographical locations for collaborative work. Many customers use SyncIQ to replace physical tape-based backup environments by protecting data to another Isilon cluster. Depending on the business requirements, SyncIQ also enables customers to realize their disaster recovery objectives.

Following are some considerations for setting up SyncIQ as a backup solution, protecting data to another Isilon cluster:

- Create policies that group several projects. For example, if the total amount of all data to be protected on the source cluster is 600TB, then instead of backing up all data in one policy, break it down to four policies.
- Do not mix categories of data in a single policy, i.e., do not configure projects and home directories to backup in the same policy, keep them separate.
- EDA Tools, binaries, library data – these may not need daily backups, as there are no daily changes, consider backing them up weekly.
- Keep the above tool areas quoted at smaller capacities due to very large file counts – this will help improve backup, replication, and recovery times as opposed to recovering all tools from one policy during recovery efforts.
- If the target Isilon cluster is dedicated to backups, and not DR, then H5600, and H500 are the best recommended nodes for the combination of space density and performance.
- Ideally, backups in semiconductor should be configured from one source cluster to one target cluster. Backing up multiple source clusters to one target cluster is fine but consider target cluster performance during failover event along with the retention periods and how target cluster would grow as multiple source clusters scale out their capacity. In case the source cluster is in about 32 nodes or larger and growing, a one to many design is recommended, where data in one cluster is backed up among 2 or more target clusters.
- If there are data sets known to have very high change rates, to the tune of deleting one million files or more daily, consider backing them up in a separate SyncIQ policy.

Following are some considerations for SyncIQ when setup to do replication of data for the purpose of distributing datasets to various design locations. An example of this would be to distribute tools and binaries from a central Isilon cluster to clusters located in remote data centers to achieve uniformity of tools versions across the company.

- Be aware that configuring snapshots on target is optional here as the purpose is to distribute tools, and maintain one copy of it in all remote locations.
- For replicating a project to another location, using the option of “whenever the source is modified” needs care. For very active project it can trigger large amount of replication, snapshot and network traffic. If this option is necessary, use the “sync-job delay” option as well.
- For project collaboration work, given that replicated target data is available only as read-only, ensure the designers understand the setup. Local copies of data using host-based utilities can be configured to provide a writable working area with periodically copied data from the read-only area.
- While data backup policies may be consuming local network bandwidth, project replications or tools distributions consume WAN bandwidth.

If the cluster is configured for following other features, please review the “Dell EMC Isilon SyncIQ: Architecture, Configuration, and Considerations” document for further guidance:

- Compliance mode
- CloudPools
- Smart Dedupe
- SyncIQ Encryption
- Hadoop TDE

5.8.2 SyncIQ Performance Considerations

Following are recommendations from the “Dell EMC Isilon SyncIQ: Architecture, Configuration, and Considerations” document, refer to this document for additional information on this subject. The link for this is listed in the References section:

- Establish reference network performance using common tools such as Secure Copy (SCP) or NFS copy from cluster to cluster. This will provide a baseline for a single thread data transfer over the existing network.
- After creating a policy and before running the policy for the first time, use the policy assessment option to see how long it takes to scan the source cluster dataset with default settings.
- Use file rate throttling to roughly control how much CPU and disk I/O SyncIQ consumes while jobs are running through the day. Remember that “target aware synchronizations” are much more CPU-intensive than regular baseline replication but they potentially yield much less network traffic if both source and cluster datasets are already seeded with similar data.
- Use IP address pools to control which nodes participate in a replication job and to avoid contention with other workflows accessing the cluster through those nodes.
- Use network throttling to control how much network bandwidth SyncIQ can consume through the day

- **Note**: Following are the defaults as of OneFS 8.0:

- A maximum of 1,000 configured policies and 50 concurrent jobs are now available.
- Maximum workers per cluster is determined by the total number of CPUs. The default is 4 * [total CPUs in the cluster]
- Maximum workers per policy is determined by the total number of nodes in the cluster. The default is 8 * [total nodes in the cluster]
- Instead of a static number of workers as in previous releases, workers are dynamically allocated to policies, based on the size of the cluster and the number of running policies. Workers from the pool are assigned to a policy when it starts, and the number of workers on a policy will change over time as individual policies start and stop. The goal is that each running policy always has an equal number (+/- 1) of the available workers assigned.
- Maximum number of target workers remains unchanged, at 100 per node

P, workers = workers on primary cluster

sworkers = workers on secondary cluster (or target cluster), fixed at maximum of 100

pworkers per node = 4 x virtual cores per node

Max no. of pworkers per cluster = # of nodes x pworkers per node

Max pworkers per SyncIQ policy = 8 x # of nodes

Example:

F800 with 8 nodes, and 20 SyncIQ policies

Max pworkers per node = 4 x 32 (16 physical cores x 2 = 32 virtual cores pernode) = 128

Max pworkers per cluster = 8 nodes x 128 = 1024

Max pworkers per policy = 8 x 8 nodes = 64

When the first policy starts, it will be assigned 64 pworkers, with the maximum pworkers on the cluster being 1024. When the second policy starts, it will be assigned 64 pworkers. Similarly, each policy will get 64 workers until the cluster limit of 1024 is reached. Thereafter, the 17th policy to start will cause a redistribution of the 1024 workers among the 17 policies. Hence each policy gets about 60 workers.

5.8.2.1 Semiconductor Considerations

For semiconductor, there are no specific recommendations to change the pworkers or sworkers. However, on the source cluster, which nodes participate in SyncIQ replication traffic is an important criteria as outlined in above list. Consider this example - If you have two node pools in the cluster and you'd like all client traffic to be handled by the faster nodes, then, not assigning those nodes (or some of them) to the SyncIQ pool may be advantageous.

For a mixed-node cluster, when the mix of different node types and quantity are added up there are too many combinations for any golden formula to be derived for optimal SyncIQ configuration. For a homogeneous cluster with only one node type, say all F800 nodes, it's really a matter of observing performance of policies with all nodes participating versus some.

However, on the secondary cluster, all nodes should be configured to participate in the SyncIQ workload, unless that cluster is designated to be a combination role of primary and secondary.

Additionally, consider the following:

- Since the sworkers max out at 100 per node, that is the maximum number of pworkers on primary clusters that could be in use at any given time. Note, the 100/node sworker limit is regardless of the node type and it can be changed if the technical support team finds it necessary.
- Only the nodes that are part of the SyncIQ related SmartConnect pool are participating in the actual SyncIQ work. However, pworker and sworker calculations shown above apply to all nodes in the cluster. The essence here is that OneFS calculates these values dynamically, the math shown is to illustrate how those calculations are derived.
- It is important to select the appropriate node types for a secondary cluster. For semiconductor, it is recommended to use H5600 or H500 or better. The other node types, namely A200, and A2000 are principally suited for cold archives.

5.8.2.2 A note on bandwidth reservation

OneFS 8.2.0 introduces bandwidth reservation. However, the administrator must change the default to the desired value as either a percentage of total available or as bits/second.

If a bandwidth reservation is not created for a policy, the bandwidth reserve is applied. The bandwidth reserve is specified as a global configuration parameter, as a percentage of the global configured bandwidth or an absolute limit in bits per second.

If a bandwidth reservation is not configured in OneFS 8.2 for a specific policy, the default bandwidth reserve is 1% of the global configured bandwidth. The default is set at this level to encourage administrators to configure the bandwidth reservation per policy. For clusters upgrading from a previous release to OneFS 8.2, it is important to note that any existing policies default to the 1% bandwidth reservation, assuming a global bandwidth reserve is not configured.

Note: The maximum available bandwidth is calculated based on all active front-end interfaces on all nodes in the cluster. One way to determine this bandwidth is to use iPerf on the cluster. Keep in mind, the utilization of maximum available bandwidth is dependent on other bandwidth consuming services on the cluster, including client access

---

5.8.2.3 SyncIQ Encryption

OneFS 8.2.0 version offers encryption capability for SyncIQ. This allows encrypting the traffic between two clusters configured for SyncIQ policies. This is a global setting on the cluster that applies encryption to all

policies. Details on this feature can be found in the OneFS administration guides as well as the SyncIQ white paper.

Consider the following for SyncIQ encryption in semiconductor CAD design environment:

- Encryption of data is generally a business requirement. For EDA, there's no specific recommendation for using this feature.
- Encryption adds minimal overhead to the transmission, but it may impact a production workflow depending on the network bandwidth, cluster resources, workflow, and policy configuration. Only after successfully testing encryption in a lab environment and collecting satisfactory measurements, may the production cluster be considered for implementing SyncIQ encryption.
- Both the source and target cluster must be upgraded and committed to OneFS 8.2, prior to enabling SyncIQ encryption.

5.8.3 Snapshots Considerations

Snapshots on the source cluster are configured for short-term recovery of data. Snapshots consume a relatively small amount of space but provide almost instantaneous recovery of data. For most customers, users are able to recover their data themselves, negating any operational efforts. SnapshotIQ creates snapshots at the directory-level instead of the volume-level, thereby providing improved granularity. There is no requirement for reserved space for snapshots in OneFS.

In semiconductor, snapshots (in conjunction with long-term data protection strategies) can be configured as per following sample:

Project data: Daily or multiple snapshots a day with retention up to 1 months

Home directories: Daily or multiple snapshots a day with retention up to 3 months

Scratch data: No snapshots

EDA tools, binaries, libraries: Daily with retention up to 1 month

Code repositories: Daily or multiple snapshots a day with retention up to 1 months

Following are some considerations when using SnapshotIQ:

- If a directory is moved, you cannot revert any snapshots of that directory which were taken prior to its move.
- Use an ordered snapshot deletion strategy where viable.
- Using SmartPools file policies, snapshots can be configured to physically reside on a different node pool than the original data. The recommendation, however, is to keep snapshots on the same tier on which they were taken.
- The default snapshot limit is 20,000 per cluster, it is recommended limiting snapshot creation to 1,024 per directory.
- Configure the cluster to take fewer snapshots, and for the snapshots to expire more quickly, so that less space will be consumed by old snapshots. Take only as many snapshots as you require, and keep them active for only as long as you need them.
- Avoid creating snapshots of directories that are already referenced by other snapshots.
- It is recommended that you do not create more than 1000 hard links per file in a snapshot to avoid performance degradation.

- Creating snapshots of directories higher on a directory tree will increase the amount of time it takes to modify the data referenced by the snapshot and require more cluster resources to manage the snapshot and the directory.
- Do not configure snapshots of /ifs, the /ifs/.ifsvar within is a system area and remains very active. Please note though, as of OneFS 8.2, the .ifsvar is excluded from all /ifs snapshots automatically – it is a configurable feature allowing disabling this exclusion.
- Configure snapshots at higher (parent) levels for each type of dataset. For example, if all home directories are under ../homes directory then configure snapshots on that directory instead of individual homes.
- Do not disable the snapshot delete job, since this prevents unused disk space from being freed and can also cause performance degradation.
- If you need to delete snapshots and there are down or smartfailed components, or the cluster is in an otherwise degraded state, contact Isilon Technical Support for assistance.
- If you intend on reverting snapshots for a directory, it is recommended that you create SnapRevert domains for those directories while the directories are empty. Creating a domain for a directory that contains less data takes less time.
- Delete snapshots in order, beginning with the oldest. Where possible, avoid deleting snapshots from the middle of a time range.
- Configure the SSD Strategy to “Use SSDs for metadata read/write acceleration” for faster snapshots deletes.
- Do not delete SyncIQ snapshots (snapshots names that start with SIQ)
- Periodically check if any snapshots are configured with no retention period (i.e., infinite retention). These snapshots are often forgotten, causing the cluster to run out of space.
- If there are data migration activities or large deletions, pay attention to snapshot disk space utilization.

5.9 Backups using NDMP

At the trailing end of the protection continuum lies traditional backup and restore—whether to tape or disk. With high RPO and RTOs, this is the bastion of any data protection strategy and usually forms the crux of a ‘data insurance policy’.

In environments using SyncIQ to replicate data from a primary source cluster to a DR target cluster, performing NDMP backups on the target cluster is preferred practice. In this way, all the resources required to run the backups (CPU, memory, I/O, network) fall to the target, freeing the primary cluster up to service client I/O. Use Backup Accelerator to FC tape libraries where possible.

5.9.1 Direct NDMP

Direct NDMP (2-way NDMP) is the most efficient model and results in the fastest transfer rates. Here, the data management application (DMA) uses NDMP over the Ethernet front-end network to communicate with the Backup Accelerator. On instruction, the Backup Accelerator, which is also the NDMP tape server, begins backing up data to one or more tape devices which are attached to it via Fibre Channel.

The Backup Accelerator is an integral part of the Isilon cluster and communicates with the other nodes in the cluster via the internal Infiniband network. The DMA, a separate server, controls the tape library’s media management. File History, the information about files and directories, is transferred from the Backup Accelerator via NDMP to the DMA, where it is maintained in a catalog.

Direct NDMP is the fasted and most efficient model for backups with OneFS and obviously requires one or more Backup Accelerator nodes to be present within a cluster.

On the Gen6 nodes (running OneFS 8.2), you could purchase a Hybrid Card that is installed inside a node. This card replaces the existing front-end card with a hybrid card that provides dual 10Gbps Ethernet ports and dual 8Gbps Fiber Channel ports for direct attachment to your tape drive. Please contact your sales team for further information around compatibility and restrictions.

Along with the hybrid card, in OneFS 8.2, there are two new features – NDMP Redirector and the NDMP Throttler. With the redirector, the NDMP daemon checks loads of other nodes when starting a new NDMP session. This includes CPU usage, number of NDMP operations already running, and availability of tape target used for the operation. Essentially it provides automatic load distribution of NDMP operations. This feature is for direct NDMP backups. The NDMP Throttler on the other hand also works with 3-way NDMP and it's function is to limit the CPU usage on the node to prevent NDMP from overwhelming it. Check the Isilon documentation for further details on these features.

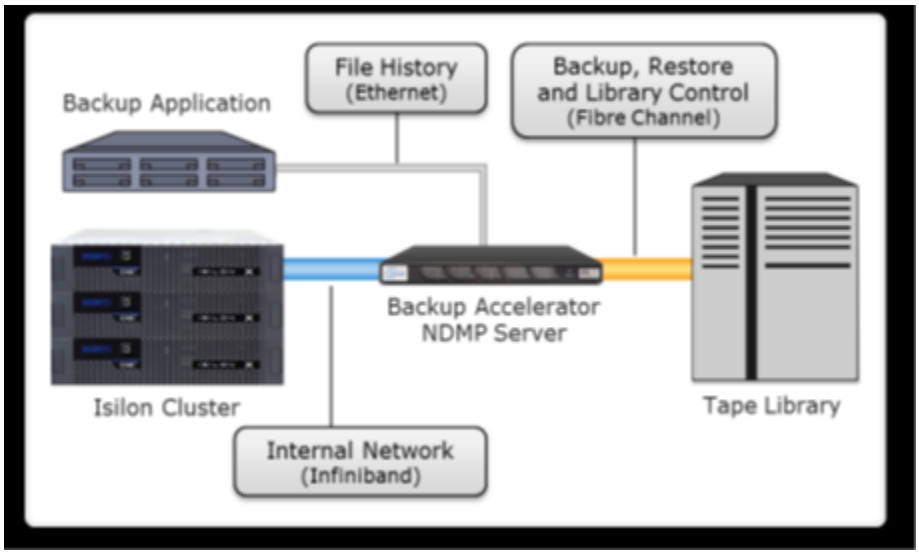

Figure 6: Recommended Two-way NDMP with Backup Accelerator

5.9.2 Remote NDMP

With remote, or 3-way, NDMP there is no Backup Accelerator present. In this case, the DMA uses NDMP over the LAN to instruct the cluster to start backing up data to the tape server - either connected via Ethernet or directly attached to the DMA host. In this model, the DMA also acts as the Backup/Media Server. During the backup, file history is transferred from the cluster via NDMP over the LAN to the backup server, where it is maintained in a catalog. In some cases, the backup application and the tape server software both reside on the same physical machine.

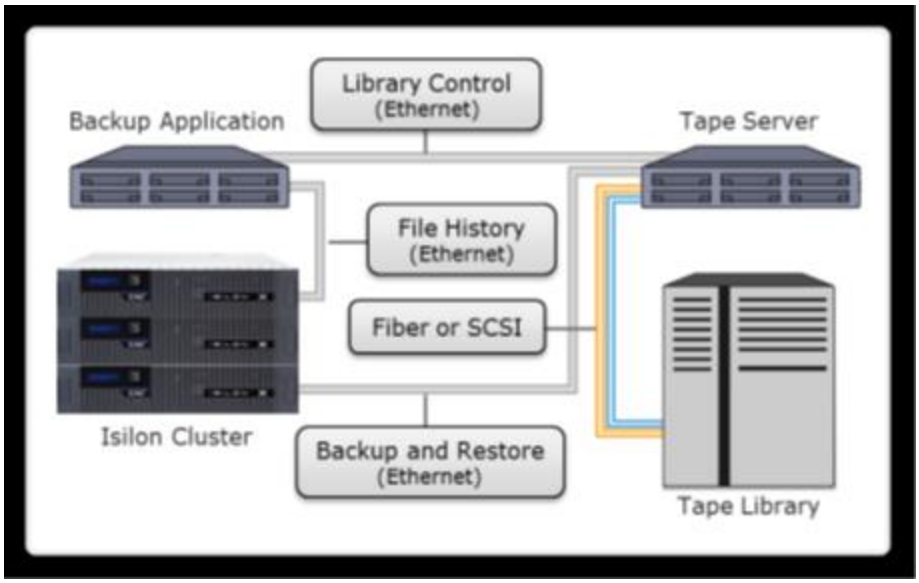

Figure 7: Remote Three-way NDMP Backup

For optimal performance, please consider the following best practices:

- The number of NDMP connections per node should not exceed 64 connections.
- NDMP to tape or virtual tape should ideally be performed on the secondary DR cluster, freeing up resources on the primary cluster to satisfy client IO.
- Where possible, use an Isilon backup accelerator node to connect to a Fibre Channel tape library or VTL. The ideal ratio is one backup accelerator per three nodes.
- If running 3-way NDMP on a primary cluster, constrain it to its own dedicated interfaces on a separate SmartConnect zone, ideally on a lower priority storage tier.
- Enable parallelism for the DMA if the DMA supports this option. This allows OneFS to back up data to multiple tape devices at the same time.
- Run a maximum of eight NDMP concurrent sessions per A100 Backup Accelerator node and four NDMP concurrent sessions per Isilon IQ Backup Accelerator node to obtain optimal throughput per session.
- NDMP backups result in very high Recovery Point Objectives (RPOs) and Recovery Time Objectives (RTOs). You can reduce your RPO and RTO by attaching one or more Backup Accelerator nodes to the cluster and then running two-way NDMP backups.
- The throughput for an Isilon cluster during the backup and recovery operations is dependent on the dataset and is considerably reduced for small files.
- If you are backing up large numbers of small files, set up a separate schedule for each directory, or high level directory.
- If you are performing NDMP three-way backups, run multiple NDMP sessions on multiple nodes in your Isilon cluster.
- Recover files through Direct Access Restore (DAR), especially if you recover files frequently. However, it is recommended that you do not use DAR to recover a full backup or a large number of files, as DAR is better suited to restoring smaller numbers of files.
- Recover files through Directory DAR (DDAR) if you recover large numbers of files frequently.
- If possible, do not include or exclude files from backup. Including or excluding files can affect backup performance, due to filtering overhead.

6 Industry Reference Design Approaches

This section covers different industry reference design approaches. Customers in semiconductor industry deploy Isilon scale-out storage in a variety of ways that encompass meeting a spectrum of business requirements. As the semiconductor design and manufacturing continues to leverage newer technologies like Artificial Intelligence (AI), Deep Learning (DL), and Machine Learning (ML), customers are rapidly enhancing their approach to shared storage deployments. Data is the new capital and asset for companies, and it is growing exponentially. Combined with vast amounts of data generated from much denser chip designs, and growing compute farms that are now introducing GPUs, the scale-out storage is a key piece in the minds of storage architects, managers, and even the chip designers and software teams. In fact, all constituents are stakeholders that influence how unstructured storage affects their ability to deliver value and improve their time to market metrics.

Following are some real customer deployment examples that may shed some light on what we may call EDA trends or reference design approaches customers are taking to solve their challenges.

6.1 Case 1: Large all-flash scratch space

This customer uses dedicated scratch storage space for some of their IC design workflows. The legacy system was unable to meet the growing size of the design projects, which were approaching several Petabytes per design. In addition, the old platform had grown into many separate siloed systems that would break the design into many partitions, increasing both design environment complexity and IT operational challenges. Since scratch space typically implies the highest levels of performance and scalability, any replacement storage solution should meet or exceed both criteria. The customer decided to use the F810 based nodes to build several ‘scratch clusters’ for allocation to different design groups across different locations. While many customers utilize a certain amount of cluster space for scratch, this customer, due to the performance and scalability requirements chose to dedicate Isilon F810 clusters just for scratch space needs. This strategy can help speed up the run times for jobs that generate lot of transient datasets that are business critical for a short period of time. Business doesn’t require protecting this data with snapshots or backups, thereby simplifying the layout, deployment, and operations, while improving the overall time to market for the designs.

In addition, the customer enjoyed significantly reduced physical floor space, power, and cooling requirements. Compared to the legacy systems, what took several racks to host the data can now be inside one rack, while delivering much better performance and allowing to grow the system to accommodate more workflows in future.

6.2 Case 2: Data protection with high-density and performance

Like many other customers across the industries, this customer is tape-less when it comes to data protection. However, as is the case with so many other semiconductor industrial customers, protecting the exponentially growing data, and increasing data center hosting costs required a much denser yet high performance solution. Customer’s Gen5 Near-Line (NL) nodes that are used as SyncIQ target cluster were replaced with the H5600 nodes. The H5600 nodes offer a unique combination of capacity scalability and performance. Each chassis provides close to 1 PB in a 4U chassis using 12TB disk drives. Combined with the in-line data reduction capabilities, it results in significant savings in rack densities while delivering much better performance compared to their previous generation NL nodes.

The added performance helps meeting performance goals with SyncIQ based backups, while significantly reducing the hosting costs as well as improving operational efficiencies.

A Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

A.1 Related resources

Isilon OneFS Design Considerations and Best Practices: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h17240_wp_isilon_onefs_nfs_design_considerations_bp.pdf

Isilon OneFS Best Practices Guide: http://www.dellemc.com/en-us/collaterals/unauth/white-papers/products/storage/h16857-wp-onefs-best-practices.pdf

Dell EMC Isilon: Network Design Considerations: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h16463-isilon-advanced-networking-fundamentals.pdf

Dell EMC Isilon SyncIQ: Architecture, Configurations, and Considerations: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h8224_replication_isilon_synciq_wp.pdf

Dell EMC Isilon OneFS Best Practices: http://www.dellemc.com/en-us/collaterals/unauth/white-papers/products/storage/h16857-wp-onefs-best-practices.pdf

Dell EMC Isilon Best Practices for Large Isilon Clusters: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h17300_wp_best_practices_large_isilon_clusters.pdf

Dell EMC Isilon: Non-Disruptive Upgrade Best Practices: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h17459_wp_isilon_non_disruptive_upgrade_ndu_best_practices.pdf

Dell EMC Isilon: Network Design Considerations: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h16463-isilon-advanced-networking-fundamentals.pdf

Isilon OneFS NFS Design Considerations and Best Practices: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h17240_wp_isilon_onefs_nfs_design_considerations_bp.pdf

STORAGE QUOTA MANAGEMENT AND PROVISIONING WITH DELL EMC ISILON SMARTQUOTAS: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h10575-wp-quota-management-with-smartquotas.pdf

ONEFS JOB ENGINE: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h12570-wp-isilon-onefs-job-engine.pdf

STORAGE TIERING WITH DELL EMC ISILON SMARTPOOLS: https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h8321-wp-smartpools-storage-tiering.pdf