PowerScale OneFS with Baselight by FilmLight

Best Practices and Configuration

March 2023

H19487

White Paper

Abstract

This white paper describes how to configure PowerScale OneFS storage for use with the Baselight by FilmLight color grading system. The latest generation of PowerScale storage nodes support 8K and high frame rate 4K playback in Baselight.

Dell Solutions

DELL Technologies

Copyright

The information in this publication is provided as is. Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Dell Inc. believes the information in this document is accurate as of its publication date. The information is subject to change without notice.

Contents

Executive summary

Overview

FilmLight Baselight is a color grading and image processing system that is widely used in cinematic production. Traditionally, Baselight DI workflows are the domain of SAN or block storage. With the latest Dell PowerScale OneFS, all-flash PowerScale nodes can support the demanding requirements of Baselight workflows including 8K and high frame rate 4K uncompressed playback.

This white paper includes details about how Baselight uses external storage and how best to connect PowerScale storage to the system. The paper also includes information about the various caching mechanisms in Baselight. When designing a storage infrastructure to support Baselight playback, understanding the interaction between the Baselight timeline, internal cache, and external storage is essential. Leveraging PowerScale support of Remote Direct Memory Access (RDMA) or nconnect, as described in this paper, is another essential step in achieving the high throughput requirements of Baselight playback.

Audience

This guide is written for media technology professionals who are planning to deploy Baselight and PowerScale OneFS. An understanding of storage terminology, Linux operating systems, and media workflows is assumed.

Revisions

Date	Description
March 2023	Initial release

We value your feedback

Dell Technologies and the authors of this document welcome your feedback on this document. Contact the Dell Technologies team by email.

Author: Gregory Shiff

Contributor: Darrin Smart

Note: For links to other PowerScale documentation, see the PowerScale Info Hub.

How Baselight uses media storage

Baselight system types

Baselight is available in several different configurations. At its most basic is the Baselight ONE System. This system in a single stand-alone workstation with powerful CPUs, GPUs, and some fast internal storage. The next step up in complexity is the Baselight TWO system. This configuration has a processing node that typically lives in the data center and a lightweight host UI system that drives the Baselight graphical interface. These two computers are connected using a private 1 GbE network managed by the Baselight system. The compute node in the data center has multiple GPU and CPU resources to do the heavyweight Baselight processing tasks, while the host UI system is small (and quiet).

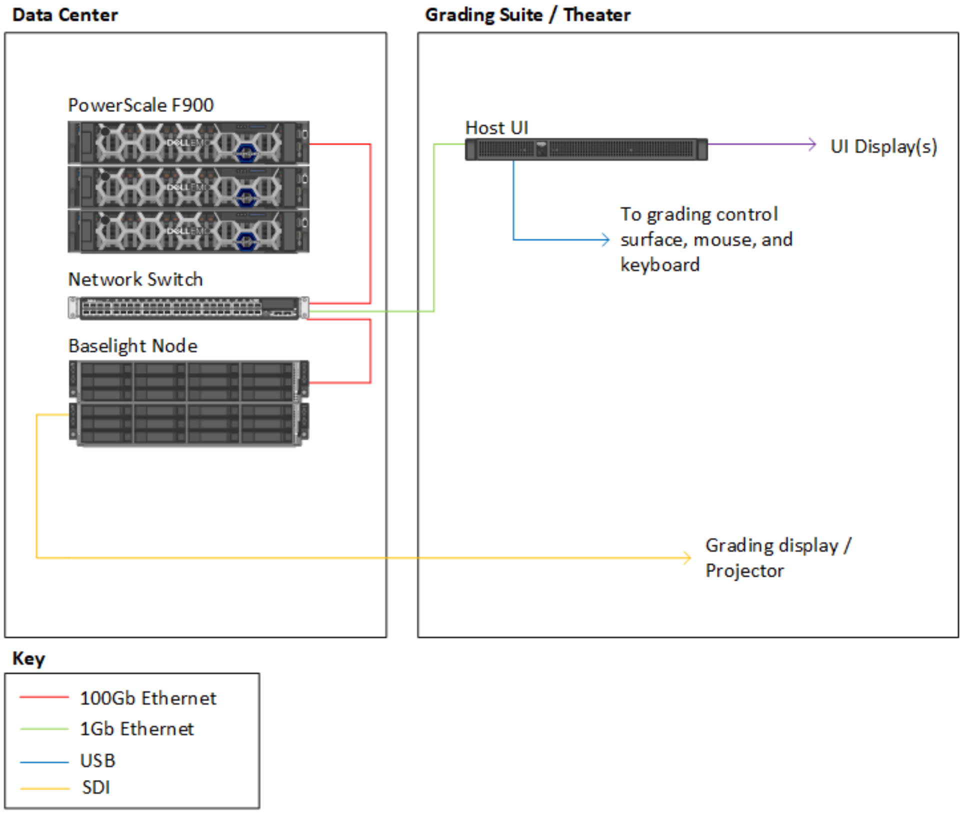


Figure 1. Baselight TWO System

The most complex Baselight system is Baselight X. Baselight X is similar to Baselight TWO but provides double the processing power and internal storage resources for the most demanding workflows. When Baselight TWO or Baselight X is used with external storage (PowerScale), both the processing nodes and the host UI node need access to that storage. The UI host only needs 1 GbE access to the external storage, whereas the processing nodes ideally use 100 GbE connections.

Caching in Baselight

Baselight makes extensive use of caching on the internal storage in both the Baselight ONE workstation and the processing nodes of Baselight TWO/X. As media is added to the Baselight timeline, that media is read from whatever storage it is originally resident on. For instance, if the media is on PowerScale storage, it is read from the PowerScale. When the Baselight operator applies processing to that media, the processed frames are rendered and stored in the local Baselight cache. During playback, Baselight switches between pre-rendered frames in cache and original media stored on PowerScale.

Another option in Baselight is to mark media for pre-caching (and pre-rendering). This option is helpful when playing back media that is especially taxing to the workstation. Examples include playback of PIZ compressed EXR and doing intensive tasks such as grain removal. The Baselight timeline is organized with “strips” of media, grades, and other image processes, forming a “stack” of operations. These grading stacks go from top to bottom. It is possible to mark one strip to be cached. All processing for that strip and the strips above it are then pre-rendered and stored in the local cache. Baselight operators must be careful with this “strip cache” feature. If each “strip” in the stack is individually marked for pre-rendering, Baselight might create a lot of unnecessary intermediate cache files. It is more efficient to pre-cache only the output of a set of strips of media.

Monitoring playback in Baselight

Baselight includes a system for monitoring media playback, which is displayed by selecting **Views** and **Playback Monitor**. Playback Monitor is a useful troubleshooting and benchmarking tool for determining how well the Baselight system is keeping up with reading media from storage and processing it in CPU and GPU.

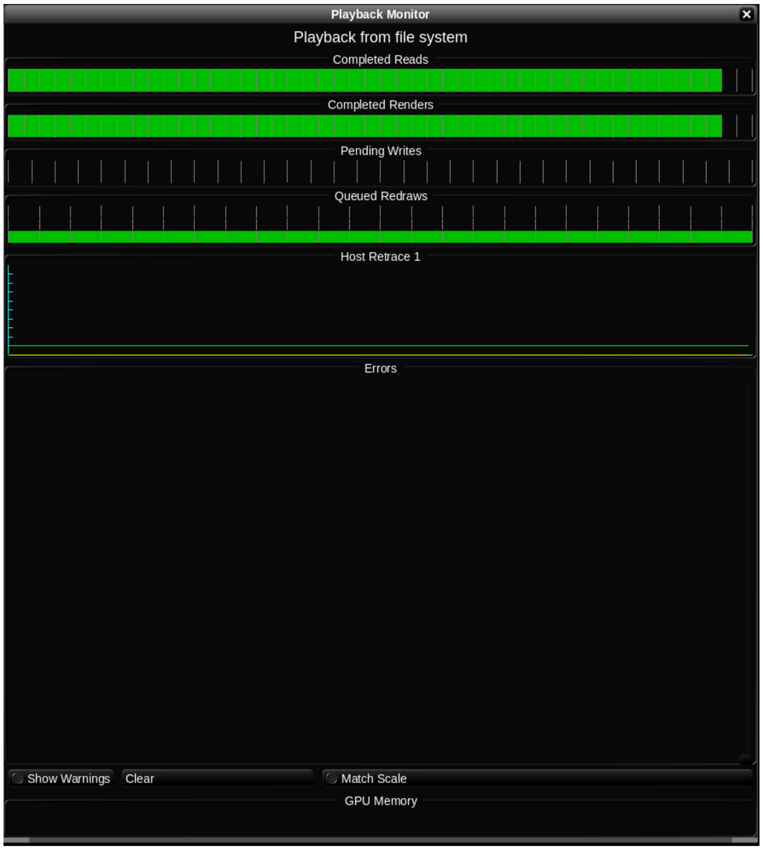


Figure 2. Baselight playback monitor

In terms of storage performance, the **Completed Reads** bar at the top of the Playback Monitor is the most significant. This bar shows that Baselight can read frames from storage fast enough while maintaining enough read-ahead buffer for smooth playback. If this line falls all the way to the left, Baselight will drop frames.

PowerScale strategies for Baselight

Mount options

Baselight ONE, Baselight TWO, and Baselight X run in a customized version of CentOS 8. Thus, the NFS protocol is the preferred method for connecting PowerScale storage to Baselight. NFS has many tuning parameters that often need to be tweaked to individual environments and workflows. However, NFS over RDMA and nconnect are two NFS connectivity strategies that greatly improve performance. Intelligent use of these options makes PowerScale an excellent choice for Baselight.

OneFS can be configured to prefetch sequentially named files. The Filename-based Prefetch feature was designed to support image-sequence-based workflows, which are common in Baselight. Using Filename-based Prefetch is another key to achieving the performance demanded by Baselight.

The following sections describe NFS over RDMA and nconnect. In our testing, RDMA yielded the best results. However, not every network can support RDMA traffic. We recommend using RDMA when the network and NIC support it; otherwise, use nconnect with TCP.

NFS over RDMA

RDMA enables data in the storage server to be transferred directly to the RAM of the storage client. This communication bypasses many buffering layers required by TCP-based connections. RDMA has two main benefits for these workflows:

- NFS over RDMA offers roughly double the performance of standard TCP-based NFS.
- NFS over RDMA greatly reduces the CPU load on both the storage server (PowerScale) and the storage client (Baselight workstation). This reduction leaves more power in the storage server to deliver data and more power in the client system to process video.

RDMA does have various requirements: a lossless network, flow control (global or priority flow control), and a supported NIC. The FilmLightOS onboard Mellanox driver has RDMA support. FilmLight does not recommend installing the Mellanox OFED drivers due to compatibility issues with SMB.

For an in-depth discussion of RDMA in media workflows, see the *PowerScale OneFS: NFS over RDMA for Media* white paper.

NFS and nconnect

RDMA is the best option when available. However, sometimes implementing RDMA is not possible due to client NIC or network constraints. In these situations, TCP-based NFS connections can be used with the nconnect mount option. The nconnect option has been back ported for use in CentOS 8 in the Linux 4 kernel and so is available to Baselight client systems.

The nconnect mount allows the client system to open multiple parallel TCP connections between the client and the PowerScale storage. The maximum and recommended nconnect value is 16. Using nconnect with TCP-based communication results in nearly the same NFS throughput as RDMA. However, nconnect does not offer the same CPU load reduction benefits.

Filename-based Prefetch in OneFS

PowerScale OneFS combines storage, volume management, and connectivity into a single system. By combining these storage layers into a cohesive whole, OneFS can deliver data to client systems with an intelligence that would otherwise not be possible. One of these technologies is Filename-based Prefetch.

When enabled, Filename-based Prefetch informs the OneFS file system to aggressively prefetch files with sequential names, such as image sequences. Filename-based Prefetch is configured in the OneFS command line and enabled on a per-directory basis. We do not recommend enabling Filename-based Prefetch on the entire file system because it can result in false prefetches.

The *Filename Based Prefetch* white paper provides a detailed description of Filename-based Prefetch and how to apply it. The paper also describes the various strategies that OneFS employs to lay out and access data on disk.

Filename-based Prefetch is required for 8K image sequence playback in Baselight.

Enabling Direct IO

While Direct IO is not an NFS mount option, it does affect storage performance. Direct IO can be configured when setting up external storage options in Baselight, as described in Setting volume mount options in FilmLightOS.

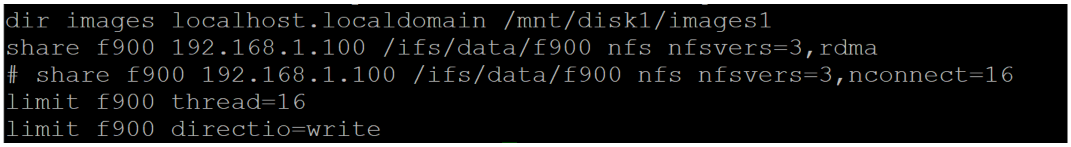
Direct IO bypasses some memory caching in Linux. In this case, with Baselight reading directly from PowerScale, testing showed that enabling Direct IO improved write performance and degraded read performance. Baselight provides configurations to use Direct IO only for writes and not reads. This option can be configured in the `/usr/fl/etc/volume.cfg` file with the `directio=write` option.

Setting volume mount options in FilmLightOS

Volume.cfg file

How Baselight mounts and interacts with network storage is configured with a special file: `/usr/fl/etc/volume.cfg`.

In that configuration file, NFS mount options and global storage handling flags can be set. Here is a sample of output from a Baselight system that is configured to use PowerScale OneFS storage:



```
dir images localhost.localdomain /mnt/disk1/images1
share f900 192.168.1.100 /ifs/data/f900 nfs nfsvers=3,rdma
# share f900 192.168.1.100 /ifs/data/f900 nfs nfsvers=3,nconnect=16
limit f900 thread=16
limit f900 directio=write
```

Figure 3. /usr/fl/etc/volume.cfg file

In this example, the share f900 is mounted with NFS version 3 and RDMA. There is a commented-out line that shows the same share mounted with nconnect threads instead of RDMA.

Thread count and Direct IO

The other notable options are configured underneath the storage and connectivity specifications, as shown in the preceding figure (Figure 3). The f900 share will use 16 I/O threads in Baselight. The thread option is indicated with the `limit f900 threads=16` line. Those threads are the number of I/O threads that Baselight will use for the volume named f900 in this example. This option holds for TCP or RDMA-based connections.

The threads setting here is distinct from the nconnect. Nconnect specifies the number of TCP channels that the operating system uses when connecting to a server with TCP. With multiple channels, the client can have more NFS requests in-flight simultaneously, increasing throughput.

The next setting tells Baselight to use Direct IO for writes only on the volume named f900.

```
limit <volume name> directio=write
```

In our testing, increasing the thread count to 16 and enabling Direct IO only for writes proved to be the best options for Baselight performance when it was reading from PowerScale OneFS.

NFSmount.conf for Baselight TWO and Baselight X

Another option for setting mount options in Baselight is the `/etc/nfsmount.conf` file. In a Baselight ONE system with only a single workstation, setting the volume mount option in `/usr/fl/etc/volume.cfg` is fine. However, Baselight TWO and Baselight X systems share a single `/usr/fl/etc/volume.cfg` file between the Baselight nodes and UI host. This shared configuration file can cause issues because the UI host NIC does not support RDMA and needs its own NFS mount options.

If NFS mount options are not specified in the `/usr/fl/etc/volume.cfg` file, the Baselight automount script will look for mount options `/etc/nfsmount.conf`. This file is unique for each Baselight node and UI host, allowing for unique mount options to be set for each system.

For technical support and configuration assistance, contact FilmLight support:

baselight-support@filmlight.ltd.uk

References

Dell Technologies documentation

The following Dell documentation provides additional and relevant information. Access to these documents depends on your login credentials. If you do not have access to a document, contact your Dell representative.

- Filename based prefetch
- NFS over RDMA for Media Workflows

FilmLight documentation

The following FilmLight documentation provides additional and relevant information:

- Baselight technical manuals