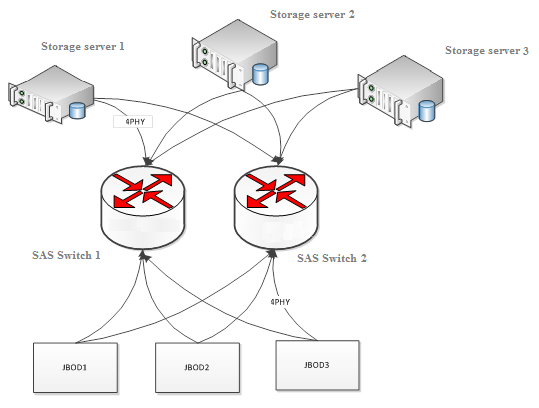
**SurFS Product Architecture**

# Hardware Architecture

SurFS storage is based on SAS (Serial Attached SCSI) switch technology to build a storage network with all the components are connected to the SAS switch 4PHY.



The system has 3 components:

1. SAS switch: offers 16 4PHY ports, each PHY has 6Gbit/s bandwidth, thus a physical port will get a 24Gbit/s bandwidth. SurFS uses double SAS switch to keep the high availability of the physical link.
2. JBOD: each JBOD could hold up to 45 SAS or SATA disks, JBOD is connected to the SAS switch by a 4PHY port, thus the 45 disks are sharing the 24 Gbit/s bandwidth.
3. Storage server: the storage servers are connected to the SAS switch through HBA cards by 4PHY too.

## SAS Switch

The SAS switch technology is the core component of the SurFS storage system; the storage servers are communicating to JBOD via the SAS switch. All the components are connected to the SAS switch by 4PHY, so the bandwidth between the SAS switch and the storage nodes is 24Gbit/s. SurFS uses two redundant SAS switches. This means there are two physical links between each storage node and each JBOD. When one physical link is disconnected, it won’t influence the other one.

## JBOD

With SurFS all mounted disks are part of the JBOD array, then connected to the SAS switch via JBOD, which means that all disks in one JBOD will share 24Gbit/s bandwidth. One JBOD could hold up to 45 disks, so each disk will get 600Mbit/s bandwidth on average if the JBOD is fully loaded. As all the disks won’t work at the same time, there will always be enough bandwidth.

## Storage Node

The SurFS management tools and the NAS server are installed on the storage server.

# Software Architecture

## Storage Server

The storage server runs on CentOS Linux with ZFS as file system. Each storage server is mounting one or more disk from the JBOD array to build a storage pool and setting up RAID, thus all the disks on the JBODS are separated by ZFS and building a local file system separately; setting different volume tags for all the file systems on different storage servers. That way the global storage pool managed by the SAS switch can be accessed as shown below:

SAS Switch

Storage Node 1

builds ZFS file system (volume 1)

Storage Node 2

builds ZFS file system (volume 2)

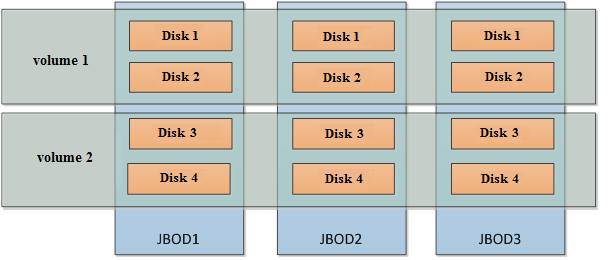
SAS PATH

SAS PATH

SAS PATH

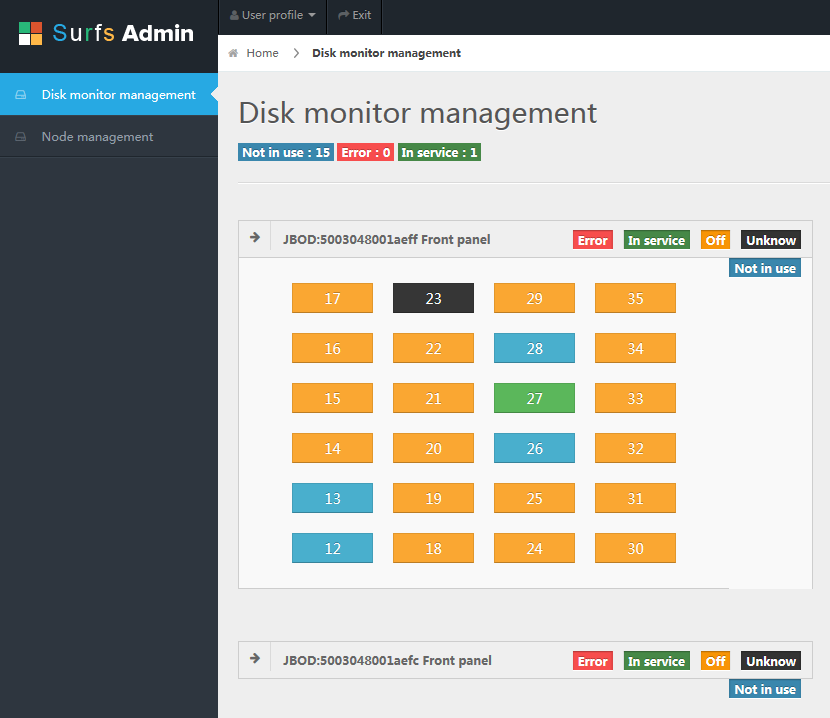
SAS PATH

SAS PATH



As shown in the figure above, server 1 has the control over volume 1 and server 2 has the control over volume 2.

SurFS offers a monitoring web UI, where users can see all disk statuses. Users also can check for status of the storage servers.



NOTE: ZFS is currently used to manage the volumes in the storage pool. The next version of SurFS will be using software RAID to instead ZFS.

Block storage service:

Client computers and the storage servers are connected by TCP/IP network.

Storage node 1

Storage service-->(access volume1)

Storage node 2

Storage service-->(access volume2)

Guest machine 1

Storage client service

Guest machine 2

Storage client service

TCP PATH

TCP PATH

TCP PATH

TCP PATH

SurFS offers a set of command line tools to manage the block devices, e.g. global configuration of SurFS block storage:

* Pool management
* Volume management
* Snapshot management
* Management of volume export

## NAS Storage Client

The NAS storage client network can be used as as block storage; the storage server supports NFS and CIFS protocols. Users can use NFS/CIFS clients to connect to the storage server.

The NAS storage offers storage nodes load balancing, when a user sends a write request, the client needs the storage to determine which node to write to, the write strategies are:

1. Round Robin
2. Nearby, finding the node with the best bandwidth. Most of time the client is a VM, thus ‘Nearby’ will be the best performance strategy.
3. Dynamic polling: the client will collect the IO loading information of all nodes, then taking the node with the lowest load to write to.

# SurFS Advantages

SurFS storage architecture takes full advantage of the SAS storage network based on SAS switch technology to build a low cost, high performance, large capacity storage system.

* Low cost:

In a SurFS storage system, the data is transferred via the SAS network, which is a large bandwidth, low latency storage network. As the four-way wide connection is using SAS-2(6Gb/s), with a 4PHY which is offering a total of 6\*4=24Gbit/s bandwidth at lower cost than a IP storage network with the same bandwidth.

* Large capacity:

In a SurFS storage system, the disks are connected to SAS storage network by JBOD with a single cluster can hold up to 400 hard disks, which makes a total capacity of 1.6PB when using 4TB hard disks (4PB capacity with 10TB hard disks)

* High performance:

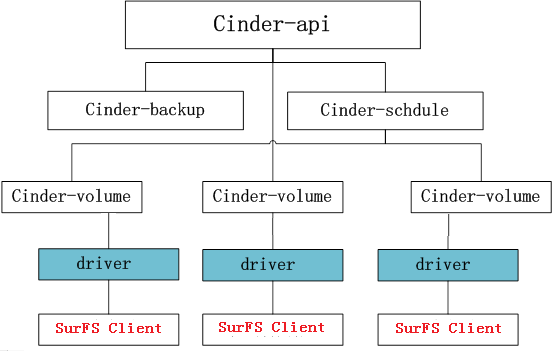
In a SurFS storage system, all the components are connected to SAS switch by 4PHY ports, means each port will get 24Gbit/s bandwidth, which is 2.4 times of 10GbE network. At the same time, since the SAS network doesn’t use the TCP/IP protocol, it has a lower latency.

* Different access way and unified storage backend:

Bases on SurFS storage, users can implement different kinds of storage services, including NAS storage service, block storage service and object storage service. Because of the unified storage backend, all of the above service could benefit from the SurFS advantages of low cost, large capacity and high performance, as well simplifying the system installation and maintenance.

# SurFS on OpenStack Use Case

SurFS’s can used as block device for Openstack, SurFS offers drivers for Cinder.



SurFS features:

|  |  |
| --- | --- |
| **Type** | **Features** |
| volume | Create volume |
| Clone |
| Volume extend |
| Delete volume |
| volume-VM | Attach to VM |
| Detach from VM |
| snapshot | Create volume snapshot for a volume |
| Create volume from snapshot |
| Delete volume |
| Image | Create volume from image |
| Create image of volume |
| volume migration | Move volume from A to B |

Users can optimize the performance by converged architecture. The VMs will access the disks directly through the SAS path to increase the IOPS and bandwidth of read/write.