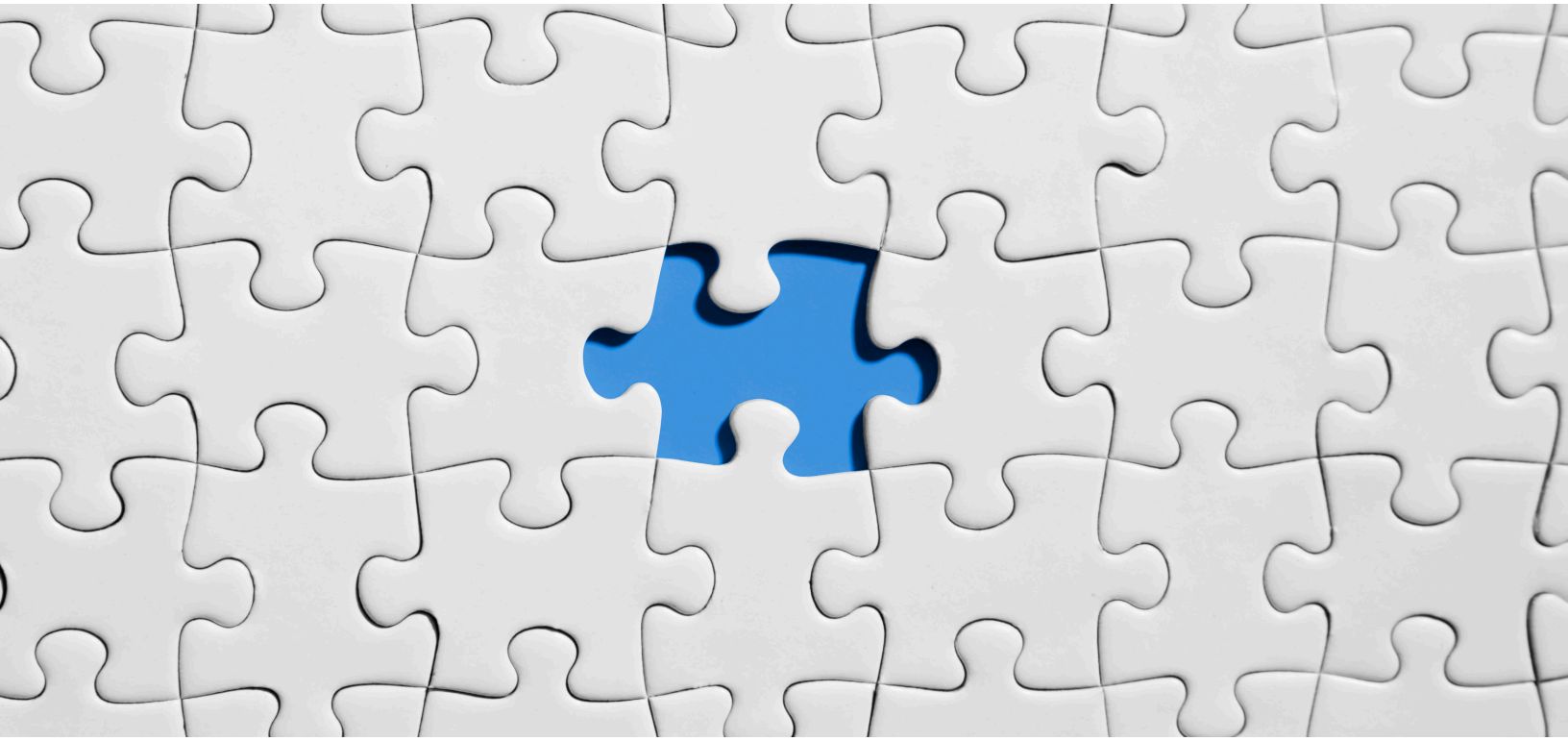
DELL Technologies

MANUAL STORAGE SIZING



Poorna Prashanth

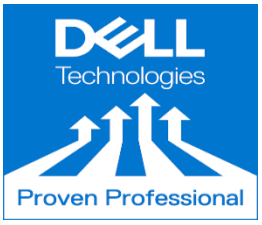
Manager

Dell Technologies

Poorna.prashanth@dell.com

Knowledge Sharing Article

DELL Technologies Proven Professional



The Dell Technologies Proven Professional Certification program validates a wide range of skills and competencies across multiple technologies and products.

From Associate, entry-level courses to Expert-level, experience-based exams, all professionals in or looking to begin a career in IT benefit from industry-leading training and certification paths from one of the world's most trusted technology partners.

Proven Professional certifications include:

- Cloud
- Converged/Hyperconverged Infrastructure
- Data Protection
- Data Science
- Networking
- Security
- Servers
- Storage
- Enterprise Architect

Courses are offered to meet different learning styles and schedules, including self-paced On Demand, remote-based Virtual Instructor-Led and in-person Classrooms.

Whether you are an experienced IT professional or just getting started, Dell Technologies Proven Professional certifications are designed to clearly signal proficiency to colleagues and employers.

Learn more at www.dell.com/certification

Table of Contents

Introduction

Understanding storage performance KPI's is important when initiating effective presales conversation with customers and to identify the root cause of performance issues which would, in turn, help better position products.

This article describes performance and capacity metrics that impact storage performance behavior and provides a methodology to perform storage sizing manually without use of any tools while considering performance and capacity metrics.

The objective of this article is to understand:

1) KPI's that drive performance of a block storage system
2) Storage Sizing methodology and procedures without the use of tools
3) Impact of certain features on overall performance, such as
    a. Workload
    b. IO Pattern/Type
    c. Tiering
    d. RAID
    e. Sparing
4) Relative dependency of performance KPI's
5) Use manual sizing methodology to size an estimate of competitive storage system

Audience

This article is intended for Dell Technologies sales and presales field personnel interested in gaining a high-level understanding of the ECS architecture and basic methodology/process/procedures to size a storage device to best meet customer requirements and understand the core metrics that impact storage performance.

Scope

This article outlines manual sizing procedures to size a storage device and to understand the significance of core performance and capacity metrics. Some of the core metrics covered are:

a) File System Alignment
b) IOPS
c) IO Type (READ/WRITE)
d) IO Size
e) IO Pattern (Sequential or Random)
f) Bandwidth
g) Capacity
h) Host Application

- Outlines the relation between core metrics
- Do's and Don't's while sizing a storage solution
- RAID types and its impact on specific workload type
- Does not discuss
    - Replication
    - Backup
    - Migration

Overall, this article provides insight to Storage Device Sizing best practices.

File System

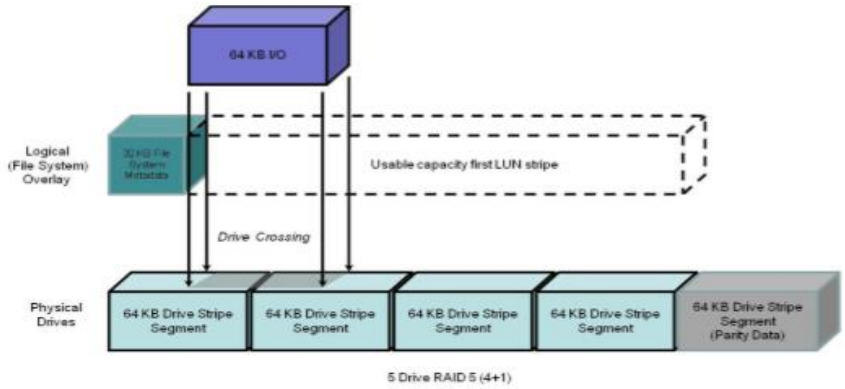
Configuring the host file system correctly can significantly enhance storage device performance. Storage can be allocated to the file system through Volume Manager. The host file system can support shared access to storage from multiple hosts.

File system caching reduces load on the file system by maximizing the storage system performance. Matching the file system I/O size with the application and the storage system has a positive impact on overall storage system performance. File system can be configured for a minimum extent size starting at 4KB and can extend to 128KB. When the goal is to move a large amount of data, larger I/O size (64KB and higher) will be positive.

File System Alignment

File System alignment or partition alignment is understood to mean proper alignment within reasonable boundaries of the data storage device. Proper partition alignment ensures ideal performance when accessing data. Misalignment leads to increased response time and reduced performance especially with SSD drives.

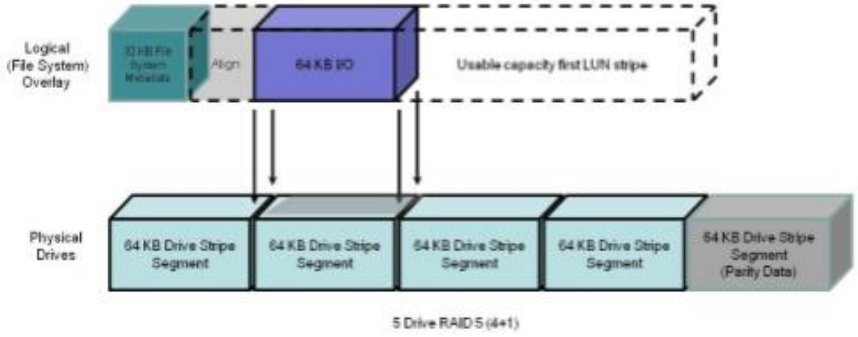
File system alignment impacts the amount of resources needed when the storage device services an IO request. A file system aligned to RAID group striping will reduce latency with increased throughput. File system misalignment adversely impacts performance and can result in drive crossings.



Knowing the IO type (Sequential, Random or mixed) and workload is important to understand the benefits of the alignment. The type and size of the data transfer is application-dependent.

Misalignment rarely affects performance with smaller IO size.

With 64KB stripe element size, all IOs larger than 64KB will involve drive crossing. To minimize drive crossing, the partition can be aligned to be on a RAID group stipe boundary.



Similarly, VMFS filesystems suffer a performance penalty when the partition is unaligned. Using the vSphere Client to create VMFS partitions avoids this problem since, beginning with ESXi 5.0, it automatically aligns VMFS3 or VMFS5 partitions along the 1MB boundary.

Workload and its significance

There is tremendous pressure to build infrastructures that are responsive to the application workload of the organization. Measure of the workload and understanding its characteristics are critical to create an optimized performing infrastructure which can satisfy the demands placed on IT by users and application owners.

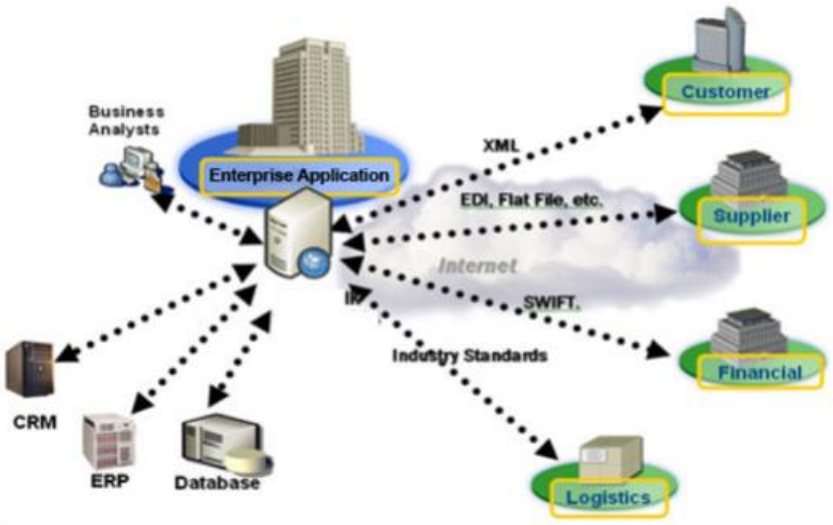
Workload is the set of input and output operations driving through data paths interfacing with storage and network infrastructures. Each workload will have unique characteristics that will impact storage latency, IOPS and throughput. These characteristics include:

- I/O Mix: Whether it is read-heavy, write-heavy or balanced

- I/O Type: Does the workload read or write data sequentially or randomly
- Block or File distribution: Does the workload write in large or small blocks
- Data Efficiency: Does the workload have highly redundant or compressible data, in which case data reduction features like compression and data deduplication works effectively

The workload shown here is aggregated workload and not a single IO. The aggregated workload is the combination of multiple workloads being generated from different applications

We can broadly categorize workloads as shown below.



Understanding workloads and its type and pattern of enterprise applications helps to design and size storage solution optimally. Workloads can be categorized into different types with a specific pattern.

Transactional Workload

Automate day-to-day business operations and processes by handling data in real time and allowing modification of existing data, removal of data and addition of new data. All data is validated in real time. These workloads are read intensive, sequential and with smaller IOs.

Examples of Transaction Enterprise Applications are SharePoint, SQL, Exchange, Oracle, and SAP.

Analytical Workload

Consolidated repository designed for reporting and analysis. Storing current and historical data, it is optimized for bulk loads and processing of large amounts of data. These systems often rely on bulk loads to load current data into the system, are read intensive and random in nature. File Servers are one example that stores large and small files from all kinds of applications. Any file saved by application and usually accessed over SMB/CIFS/NFS protocol.

High Performance Workload

These workloads are complex and require enhanced compute capabilities. Well suited for public clouds with enhanced performance, these workloads are mainly unstructured and random, mixed IO. Virtualization can be one the examples which consolidates different physical devices onto virtual environment changes the IO data pattern from and to the storage (i.e. VMware vSphere, Microsoft Hyper-V, VMware Horizon, Open stack, Cloud stack).

Data Base and Backup Workload

These workloads are huge and the performance demands require a sophisticated approach. These workloads predominantly read intensive, highly sequential and large IO's.

Data Base and Backup workloads require high speed copy and restore to minimize the impact of failures, disaster or human errors. Data can then be managed effectively for retention and long-term retrieval.

IO Type

Widely used as workload type, an IO can be a read or a write. It is the total bytes of data either ready or written from/to the storage device.

IO Size

IO Size is the average size of an aggregated workload. Typically, the IO size will be segregated as a Read IO Size and Write IO Size. IO size can range from 512 bytes to 256KB or even 1MB for more recent devices**.**

IO Size = Throughput in MB/sec * 1024/IOPS
Read IO Size = Read in MB/sec*1024/Read IOPS
Write IO Size = Write in MB/sec*1024/Write IOPS

Bandwidth

Bandwidth is quantified as the amount of data transferred in time, typically measured in bits per second. It represents the maximum data rate a link/data channel/port can transfer. It is important to ensure that configured storage would support the required bandwidth. Also known as throughput, it is measured in MB/s, which is the amount of data transfer rate. The higher the throughput, the greater the amount of data processed per second.

Throughput in MB/sec = (IOPS*IO Size)/1024

Host Application

The number of host applications, design, application configuration and modes of execution determine the behavior of the host. Enterprise applications like SQL, SAPHANA, Oracle and Exchange have integrated performance and availability features to tune application performance to the maximum. Some of these features can be used to locate bottlenecks within the system to fine-tune application performance.

It is critical to understand different IO types to correlate the behavior of the application with the server. Operational design of the host's applications affects storage system performance.

The I/O generated by application has the following broad characteristics:

- Large Block vs Small Block
- Sequential vs Random
- High Locality vs Low Locality
- Writes vs Reads
- Multi-thread vs single thread

Large Block vs Small Block

It is important to know the majority IO size and distribution of the IO size that will be in use by applications. Through this we can suggest best practices for the RAID group, Cache configuration and LUN provisioning to apply to the workload. With larger block size becoming more common of late, the definition of small and large block size has changed. Up to 16KB can be considered as small block and anything more than 64KB can be considered as large block.

Application Type	Block Size (in KB)
Data Base (Transaction Processing)	8KB
Mail Server	8KB
File Copy (SMB)	64KB
Database Log File	64KB
Web Server	64KB
Backup	64KB
Restore	64KB

Run PerfMon and Windows and IOSTAT for Linux systems to find average sectors. Multiply this number by 512bytes to get the IO Size. Large IO's deliver better bandwidth than small IOs. The use of smaller or larger block size is typically application-dependent. The decision to use large IO or fragment into small IO's requires reconfiguration at the application level, host OS, HBA and storage system LUNs.

IO Size = Throughput in MB/sec *1024/IOPS

Sequential vs Random

Knowing the type of IO is critical to predict device performance. An application can have three IO types.

- Sequential IO
- Random IO
- Mixed IO

Sequential IO are the data sets that are spaced sequentially in an addressable space while Random IOs are randomly distributed. Accessing sequential data is much faster compared to accessing random data. This means throughput with random will be much lower compared to sequential. Small Random IO's use more storage system resources than large Sequential IOs. Applications that perform only sequential IO's have better bandwidth than applications performing random or mixed IO. However, all flash is very efficient in handling random small IO's.

High Locality vs Low Locality

It is important to know the workload locality when planning for secondary caching and storage tiering. This applies only for a mechanical drive. Currently, All Flash drive technology has become the norm, and hence locality of reference might not play a big role.

Locality is based on the date set's locality of reference. Locality reference means storage locations being frequently accessed. Data that is located near each other on sector and tracks will have the highest locality. There are 2 types of locality "when written" and "where written"

- When Written: An application is more likely to access recent data than access data that was written a couple of days prior (this will be an application when deduplication is not enabled)
- Where Written: This refers to where the in-use data is distributed within its address space

A dataset workload with high locality of reference gets the best performance with secondary caching and storage tiering. This varies from application to application.

Writes vs Reads

An IO can be Read or Write. Knowing the ratio of read and write in IOs helps determine the best practice for cache, RAID group, and LUN Provisioning to apply to your workload.

Writes consume more storage systems resources than reads. Writes go to storage system write cache, is mirrored to both storage processor and eventually sent to the storage device via the backend. When writing to a RAID group, mirrored or parity data protection techniques consume additional time and resources.

Reads consume fewer storage resources than writes. Reads that find the data in cache (known as cache hit) will have faster response time than reads which needs to be fetched from a drive when it is not found in the cache (cache miss).

Multi-Thread vs Single Thread

When there are more IOs, drives become busy and IO begins to queue which increases response time. The way IO are serviced depends on threading model. The thread is the channel through which the data propagates and is serviced. The response time can be reduced by servicing requests through multiple channels concurrently. Concurrency is a way to achieve higher performance by engaging multiple drives on the storage system.

Single Threaded access means only one thread can perform IO to storage at a time. Multi-thread means 2 or more threads perform IO to storage at the same time. IO from applications becomes concurrently parallelized, which results in higher performance

RAID Groups

RAID Groups are the primary logical device in the storage system. Each RAID level has its own resource utilization, performance and data protection characteristics. For specific workloads, a RAID level can offer clear performance advantages over others.

Different RAID have different levels of capacity utilization. The difference between the data drives and parity drives is important to understand while provisioning storage. The percentage of drive capacity in a RAID Group dedicated for data decreases as the number of drives in a parity group increases. Creating large RAID groups is the most efficient use of capacity.

RAID Group Level and Size	Usable Storage
RAID 1/0 (4+4)	50%
RAID 5 (4+1)	80%
RAID 5 (6+1)	86%
RAID 5 (8+1)	88%
RAID 6	75%

The best RAID Group write performance can be achieved from the full stripe write. This is the most efficient movement of data from cache to the storage devices. Stripe Capacity is calculated as the number of user drives in RAID multiplied by block size. The default RAID Group stripe block size is 64KB.

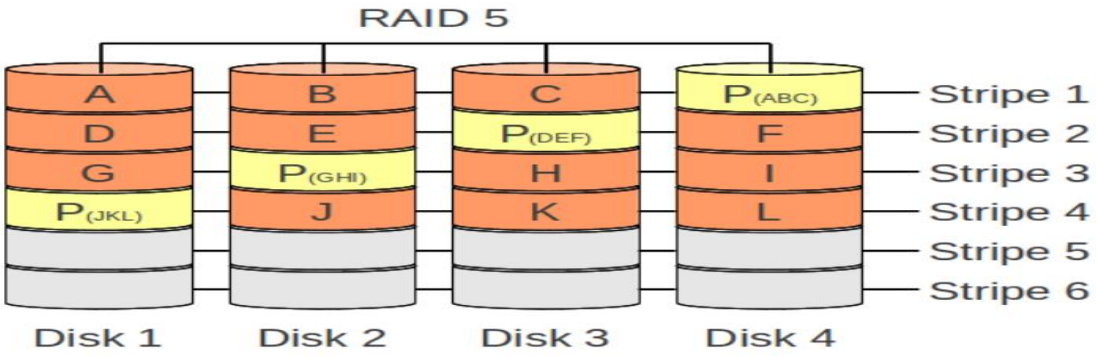
For RAID 5 (4+1) the stripe size will be 256KB (4*64). If the majority of aligned block size in not 256KB, it would result in striping to the next drive. This is less optimal for high bandwidth, large block random IO.

When to use RAID 5

RAID 5 – striping with parity – is suitable for random read performance. Random write performance is slower than read performance due to parity calculations. Sequential write and read performance is good. Sequential write performance optimizes when full stripe writes occur.

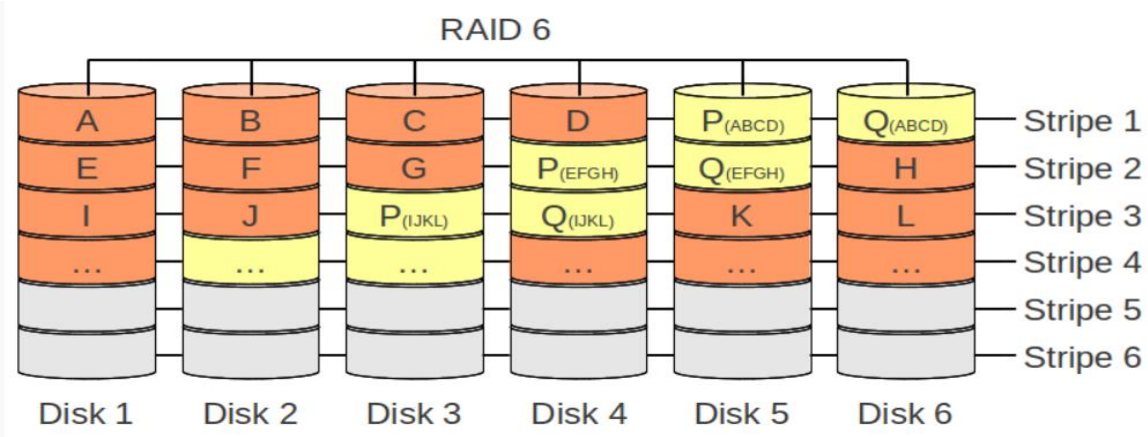
There will be additional overhead during writes to the RAID group. This is called write penalty. The write penalty of RAID 5 is 4, meaning every write to the RAID will result in 4 writes. One Write to RAID will result in below operations:

- Read the old data
- Read the old parity
- Write the new data
- Write the new parity



When to use RAID 6

RAID 6 is data striping with dual Parity. RAID 6 has similar performance as RAID 5. Random write performance is slower due to additional parity calculations. It has excellent random and sequential read performance. Performance improves with smaller stripe widths. Sequential write performance is optimized when full stripe writes occur. RAID 6 is recommended for high capacity drives and mainly preferred for capacity over performance. With the same number of data drives, RAID 5 and RAID 6 perform similar random reads while RAID 6 suffers for random writes due to additional write penalty overhead. Refer to the table below for write penalty for different RAID types



The write penalty for RAID 6 will be 6. That means every write to RAID 6 results in 6 operations in the backend as below,

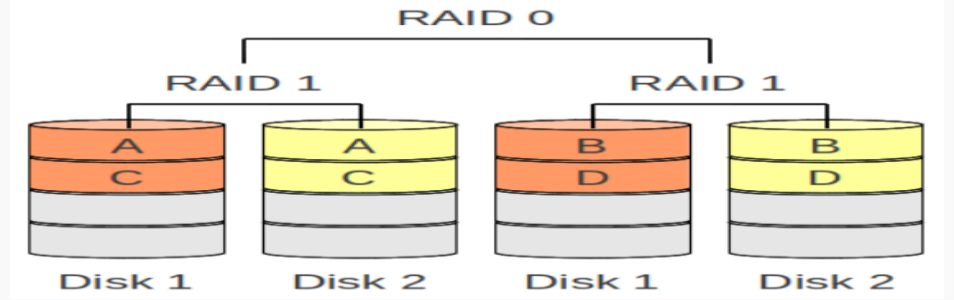
- Read the old data
- Read the old parity 1
- Read the old parity 2
- Write the new data
- Write the new parity 1
- Write the new parity 2

When to use RAID 1/0

RAID 1/0 is mirroring. It receives a performance benefit from mirrored striping and has very good random and write performance. RAID 1/0 offers better small block write performances compared to

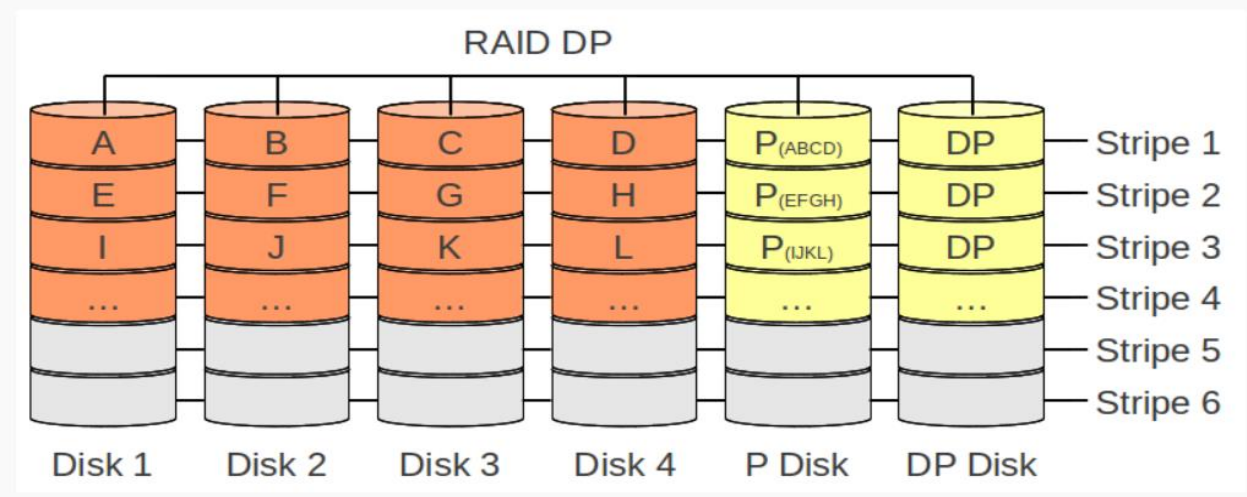
other RAIDs. While the best performance can be achieved for sequential reads, sequential writes suffer because it must write the same copy twice. RAID 1/0 offers lowest capacity utilization.

The RAID penalty for RAID 1/0 is 2.



Write Penalty for RAID-DP

Performance of RAID-DP is better than RAID 5 but can change based on the workload. Write penalty for RAID-DP is 2 because Write Anywhere File Layout (WAFL) turns incoming random IOs into sequential full stripes. During low utilization, WAFL collects logical random writes and turns them into physical full sequential writes, as a result of which it calculates the parity in the memory on the fly, that means write penalty maximum is 2. During high utilization, WAFL read the on disk blocks to recalculate the parity and then write the new stripe, this means the write penalty during high utilization increases to 4.



Dedicated and Partitioned RAID

RAID group that is aligned to one LUN using all the usable capacity is called dedicated RAID. Due to large capacity of individual drives, a RAID group can have a large capacity. Multiple LUNs can be used to partition a RAID group into smaller fragments. A RAID having more than one LUN is a partitioned RAID. LUN on partitioned RAID group consists of contagious sets of RAID stripes. Best practice is to limit the number of LUNs per RAID group to the smallest number possible to avoid possible linked contention and drive contention.

Raw IOPS = Disk Speed IOPS * Number of disks

Functional IOPS = (Raw IOPS * Write % / RAID Penalty) + (RAW IOPS * Read %)

Storage Sizing

Understanding the workload is the essential thing we need to know for storage system configurations. Workload requirements can be defined as Performance and Capacity. There are broadly two steps for storage sizing.

1. Calculate the drives required to meet the performance need
2. Calculate the drives required to meet the capacity need

Right sizing for performance can be carried out in two sub steps

1. Calculate required drives for IOPS and throughput
2. Right model the storage system to support the drive performance

In addition, you should plan for future growth. It is important to size sufficient storage to satisfy capacity and performance during peak and to meet requirements in the near future.

Sizing based on Performance

Forecasting performance is a technique that requires an understanding of the storage environment. The items below play an important role in gauging performance requirements:

- IO characteristics
- Number of IOPS
- Throughput (Bandwidth)
- Locality of reference (for hybrid drives)

Steps to obtain a performance estimate:

- Determine the Frontend IOPS
- Determine the Drive IOPS (Backend IOPS)
- Determine the number of drives to meet performance
- Determine the number of drives to meet capacity

Determine the Frontend IOPS

Determining the existing load is crucial to size new storage. Often, the workload is unknown, but it is significant to make an estimate. The estimate should include:

- Total workload (IOPS)
- Average IO Size
- Throughput
- Read% and Write %
- Capacity

Determine total IOPS and IO characteristics of a device by running the management software. Or, for Windows service. we can PERFMON and run IOSTAT for Linux systems to arrive at
Bandwidth = ((Read Size+Write Size) *IOPS).

Determine the Drive IOPS (Backend IOPS)

The drive IOPS for specific RAID can be calculated using the below formulae,

Drive IOPS = (Read IOPS + (Write Penalty*Write IOPS))
Drive MBPS= (Read MBps + Write MBps*(1+(1number of data drives in RAID group)))

Determine the number of drives to meet performance

Divide the total IOPS by per drive IOPS to get the approximate number of drives needed to handle the estimated workload. If Random IO with IO Size is larger than 16KB but less than 64KB, increase the drive count by 20%. Random IO Size greater than 64KB should meet bandwidth requirements, too.

Total Number of Drives = Total Drive IOPS/Per Drive IOPS
For example: For flash drives, Total Number of Drives = Total Drive IOPS/3500

Per Drives IOPS for different IOPS are as below,

Drive Type	IOPS
SAS 15K RPM	180
SAS 10K RPM	150
NL-SAS 7.2K RPM	90
FLASH DRIVE	3500

Per Drive IOPS for a flash drive falls drastically for random IOPS if IO size is greater than 16KB. The number of threads can increase the IOPS that can be handled. If an application is doing 4KB Random reads from flash drive, it will achieve 3000 IOPS. The same application reading a drive through 10 threads of the same size can achieve 35000 IOPS. Flash drives are well suited to multi-threading.

Determine the number of drives to meet capacity

If Data reduction features are used in the current system, effective capacity needs to be converted to usable capacity.

Usable Capacity = Effective Capacity*Data Reduction factor
Example: If data reduction of 3:1 is being achieved, the Usable Capacity = Effective Capacity (TB)*3/1

Calculate the number of drives required to meet the storage capacity requirement (usable). Add required capacity for virtual provision pool to host pool file system, which is the pool's metadata. Consider one spare for every 30 drives. Do not consider spare and system drives while calculating required drives during calculating operational performance. Mostly, total drives required for capacity will be much less than drives required to meet overall performance.

Determine the Storage Systems

Once the number of drives is estimated. Storage systems maximum supported IOPS, Maximum supported capacity, maximum supported drive count and maximum supported DAE's to host total required drives.

Storage System = Approximate Drives/Storage System Drive Maximum Count
The final number of drives is based on number of drives to meet both capacity and performance. Always consider the peak workload while determining the drives to meet performance.

The entire procedure applies to determining storage with a homogeneous pool.

Conclusion

Manual sizing of a storage device comes in handy when trying to understand the metrics that impact the performance and capacity of the storage system and how each metric is relatively dependent. The methodology documented in this article will give Sales and Presales personnel an idea of how to quickly calculate performance metrics based on application to help drive their sales/presales opportunities.