ENTERPRISE STORAGE STACK

White Paper:

Producing High Performance, Low Cost Mass Storage with ESS

Executive Overview: High performance, Flash Memory based, mass storage has historically been extremely expensive relative to spinning disks. Enterprise Storage Stack (ESS) radically reduces the cost of building a user addressable terabyte of high performance storage while actually increasing performance and durability of the storage. This white paper explains the unique methods which ESS uses to achieve this result.

Flash memory is non-volatile solid state memory which can be used as mass storage for both personal devices and for servers and appliances. Flash memory is most remarkable for its responsiveness. For instance, systems built with PCIe cards to manage Flash memory can typically random read up to 25,000 IOs per second, single thread, with a latency of just 40 microseconds between reads. Multi-thread, the same device can read at hundreds of thousands of IOs per second (IOPS). These values should be contrasted with a 7200 rpm hard disk which typically can only random read and write at about 100 IOPS.

Historically, Flash has had three problems which slowed its growth. The first was that it random wrote much more slowly than it random read. Seven or eight years ago, most Flash mass storage devices could random read at more than 2,000 IOPS, but wrote at less than 40 IOPS. The inverse of the write performance problem was that primitive update mechanisms led to rapid wear out of the devices, and the need to use parts with very high durability. These and other issues combined to make Flash extremely expensive vis-à-vis hard disk media. Just six or seven years ago, even consumer grade Flash SSDs cost over $10 per gigabyte.

Over the last few years, some of those issues have changed significantly. For instance, Consumer grade SSDs now cost around 40 cents a gigabyte, a 96% price reduction. The price of Enterprise media has also significantly declined. Similarly, the performance problem has greatly improved. For instance, current generation SSDs typically have random read performance typically in the 80,000 range, and random write in the 15,000 to 50,000 IOPS range.

On the other hand, the problem of media wear at the drive level has still not been fully tackled. Consumer media typically has wear amplification of 8:1 while many Enterprise grade drives are designed with the expectation of a 3:1 wear amplification.

Patented Enterprise Storage Stack is a transparent block filter which translates clusters of random writes into long linear atomic writes bounded on Raid-stripe and erase-block boundaries. ESS software profoundly improves the random write performance of Flash media while extending the useful life of Flash media as much as eighteen-fold in Enterprise settings. It concurrently reduces the cost of storage by allowing high performance, high durability use of Raid-5 and Raid-6 storage methods. Optional real-time block compression and deduplication can further reduce the

costs of storage and enhance media life. Optional High Availability software assures continued operation even in the presence of significant hardware failure.

Why is ESS an important technology? ESS is the only Flash management technology designed to manage flash from the outside – from the system level – as a block device. Because ESS manages from the outside, it can manage sets of SSDs, rather than just individual SSDs. Accordingly, ESS can optimize Raid performance and efficiency. Similarly, ESS can optimize Flash wear in ways not possible for internal Flash Translation Layers (FTLs). ESS can graft on functionalities such as Compression and Deduplication, and use these to increase usable storage rather than only reduce wear amplification. ESS also grafts on data reliability techniques not available to ordinary FTLs. But as important, ESS is transparent and easy to implement. ESS installs as a block filter in most versions of Linux. Once installed, it is typically out of sight and out of mind.

The durability problem of Flash memory: All flash memory has a limited number of erase cycles (or overwrites) which it can accept. The number of reliable overwrites depends upon how the flash is made. The flash used in SD cards and USB sticks typically has far less than 1,000 erase cycles of useful life. Consumer grade flash used in personal computers and some higher-end tablets typically has a life of at least 3,000 overwrites, though some products have more and some have less. Flash devices built for use in servers or storage appliances typically are expected to have a durability of 20,000 erase cycles unless some intervening technology attains the same durability without the actual use of such high-endurance Flash. Devices made with such Flash or alternate technology is generally referred to as Enterprise grade storage in the field.

The number of erase cycles in each case is not a problem in and of itself. For instance, 3,000 cycle endurance Flash represents the ability to overwrite a device 1.64 times a day for five years, while 20,000 endurance Flash represents the ability to overwrite almost 11 times a day.

Rather, the problem is due to the limit of the FTLs (Flash Translation Layer) within each solid state storage device. FTLs are designed to minimize and balance the wear applied to each erase block. As such, an FTL must manage both long linear writes and random writes as quickly and efficiently as possible.

FTLs typically perform long linear writes to SSDs in a highly efficient manner with no wear amplification at any level. Accordingly, someone such as a video production company with its gigabyte sized files and low overwrite rates can gain all the advantages of Flash while using the least expensive flash possible. This simplicity and economy of use is not true in more conventional usages where a great deal of random writing to data sets is required. Here, there are three types of wear amplification.

The first type of wear is referred to as wear amplification and is something that happens in the FTL of each solid state device. As the market has evolved, there have come to be two basic levels of wear amplification, based upon the amount of dedicated free space available on the SSD.

Consumer SSDs are designed to be as inexpensive as possible. Accordingly, they are typically built with only 8% to 13% free space. When such SSDs are almost full, they need to go through a process called defragmentation which creates new free space by taking fragmented blocks and consolidating these into totally free blocks as well as totally full blocks. With 13% free space, the defragmentation process will, on averages, build one block of free space for every seven blocks made absolutely full. As a result of all the consolidation work required in defragmentation, the wear amplification of these drives is most commonly in the 8:1 range, and a little more in some cases.

Enterprise SSDs, conversely, are typically made with much more free space. Typically, such a device might have 800 billion bytes of such storage visible, while being built from 1TB of actual flash memory. Such a device would have about 299 billion bytes of free space, or about 29% free space. The increased proportion of free space in Enterprise media radically reduces

defragmentation ratios, and accordingly wear amplification is typically reduced to a ratio of somewhere between 2:1 and 3:1.

In an Enterprise setting, the second type of wear amplification is best referred to as Raid amplification. When one is performing random writes, the Raid redundancy drive or parity stripes need to be calculated and written-to during each random write. Accordingly, anything written Raid-10 or Raid-5 will result in two writes rather than one. Anything written Raid-6 will result in three writes rather than one.

The third form of wear amplification is best referred to as write amplification. Write amplification is most commonly found in file systems. For instance, ZFS performs 11 writes per physical write for its Z1 version of Raid-5 storage, and 22 writes per physical drive for its three parity drive Z3 version of storage.

Whether the wear amplification is caused by the FTL, by Raid, or by the file system, such writing increases the wear on Flash media multiplicatively, and reduces the amount of data which can be processed at any one time.

How ESS addresses the durability problem of Flash memory. ESS uses a number of procedures to reduce the various forms of wear amplification.

ESS begins by virtualizing all white space, whether this space is mapped explicitly or created implicitly by TRIM(). Most commonly, white space (4KB blocks, all hex 00 or FF) is present in space unassigned to particular files. Given that almost all Linux file systems need 15% to 20% logical free space to keep them from falling over with congestion, this double-duty design approach creates the base-level free space needed for efficient free space defragmentation. As we will see below in the examples of the Samsung SM843T and SM843TN, this allows us to operate both with drives which have high levels of dedicated free space as well as those which have only limited free space. Virtualization can result in an effective increase in usable space of 15% to 20% without incremental costs.

ESS writes all data linearly, as atomic clusters of random writes and their associated metadata. This is extremely important in reducing wear amplification because it takes a large number of “hot spots” on the storage device and concentrates these in a single “hot spot” location. What determines wear amplification is not the average ratio of free space to work space. Rather, it is the ratio of free space to consumed space in those blocks which are actually defragmented at the time they are defragmented. Blocks that aggregate the contents of many hot spots empty much faster than scattered targets and as such tend to have very low wear amplification factors.

In ESS, virtualization and linearization together typically result in wear amplification rates below 1.3:1. This is far below the norm of most Enterprise media, which is designed for a wear amplification of between 2:1 and 3:1. The 1.3:1 value will hold true whether one is using a drive which has high levels of free space and thus amplification of 3:1 or low levels of free space and hence a typical wear amplification rate of 8:1. The reason here is two-fold. First, white space virtualization creates a high level of practical free space in its own right and this free space does not depend upon the drive hardware level free space at all. More important, the creation of a single progressing “hot spot” creates an environment where write blocks are rapidly emptied. Indeed, a wear amplification of only 1.3:1 implies that erase blocks are at least 75% empty of current data at the time they are defragmented.

Next, we need to consider the impact of Raid in wear amplification. There are several different forms of Raid. Raid-1 will always mirror data to a second drive creating Raid-amplification of 2:1 in all cases whether one is writing in a truly random manner or purely linear manner. However, Raid-5 and Raid-6 are different. If Raid-5 is written to randomly, then a parity block must be written for each random change to a raid stripe, resulting in a 2:1 Raid-amplification. Similarly, if one is using Raid-6 is a random writing context (as is wise in arrays with large drive counts), one will need two parity writes for each data write, resulting in a Raid-amplification of 3:1.

Conversely, if one is writing in a purely linear fashion, and as such writing to an entire Raid-stripe at once, wear in a Raid-5 set will be reduced from 2:1 to $1+(1/(n-1)):1$ and Raid-6 will be reduced from 3:1 to $1+(2/(n-2)):1$.

Reduction of wear-amplification together with reduction of Raid-amplification through use of ESS modalities can improve the overall durability of drives by a significant factor. In high free-space Enterprise drives, the composite durability gain will typically amount to a four- to seven-fold gain in typical relative durability. Similarly, in low free-space drives, whether Enterprise or Commercial, composite durability will typically increase by a factor of twelve to eighteen.

Further Durability Enhancers of ESS. The enhanced version of ESS also offers both high speed block compression and high speed block deduplication. Each can result either in significant durability gains, or in a significant growth in logically addressable space, or some mix of both.

When block level compression is applied to a given surface, it reduces the amount of defragmentable space by the compression ratio thus proportionately reducing wear. This is what occurs when compression is applied to media of a fixed size, such as individual devices.

Conversely, when additional logical addresses are added to the target space, and supported by adequate RAM or virtual memory, the total available logical space can grow proportionately without reducing relative free space, and thus without increasing wear. Such a mechanism works well in a managed Raid environment because any inconsistencies in compressibility are smoothed out over the whole array. It doesn't work from a drive level perspective however, as the compressibility ratios will tend to deviate from drive to drive.

When block level deduplication is applied to a given surface, any duplicates are virtually stored through referencing. As with white space virtualization, deduplication directly reduces average used space. Accordingly, wear amplification is reduced as the proportion of virtualized space grows. The question is what to do with this space?

In a small scale environment, such as individual drives, deduplication can only reduce wear because duplicates may mal-distribute. But in a large scale environment, such as a Raid-set, one can merely thinly provision logical space, and accordingly increase the total amount of space logically addressable, as long as the total amount of physically used space does not exceed safe levels. In some environments such as VDI, virtual systems, and backup, duplicated space and white space can represent 90% or even 95% of the data stored on a system due to the duplication of operating systems and white space. Accordingly, efficient memory or virtual memory solutions are essential to achieve maximum actual storage.

How many overwrites a day do you need? In the last several sections, we have discussed how Raid-topology and FTL methods impart wear-amplification as well as ways in which ESS reduces wear amplification. In the following table of overwrite factors (consolidated from data to be reported later in this document), we have taken four typical current generation Flash SSDs and extrapolated their predictable wear amplifications into net daily safe overwrite levels per gigabyte of user accessible space.

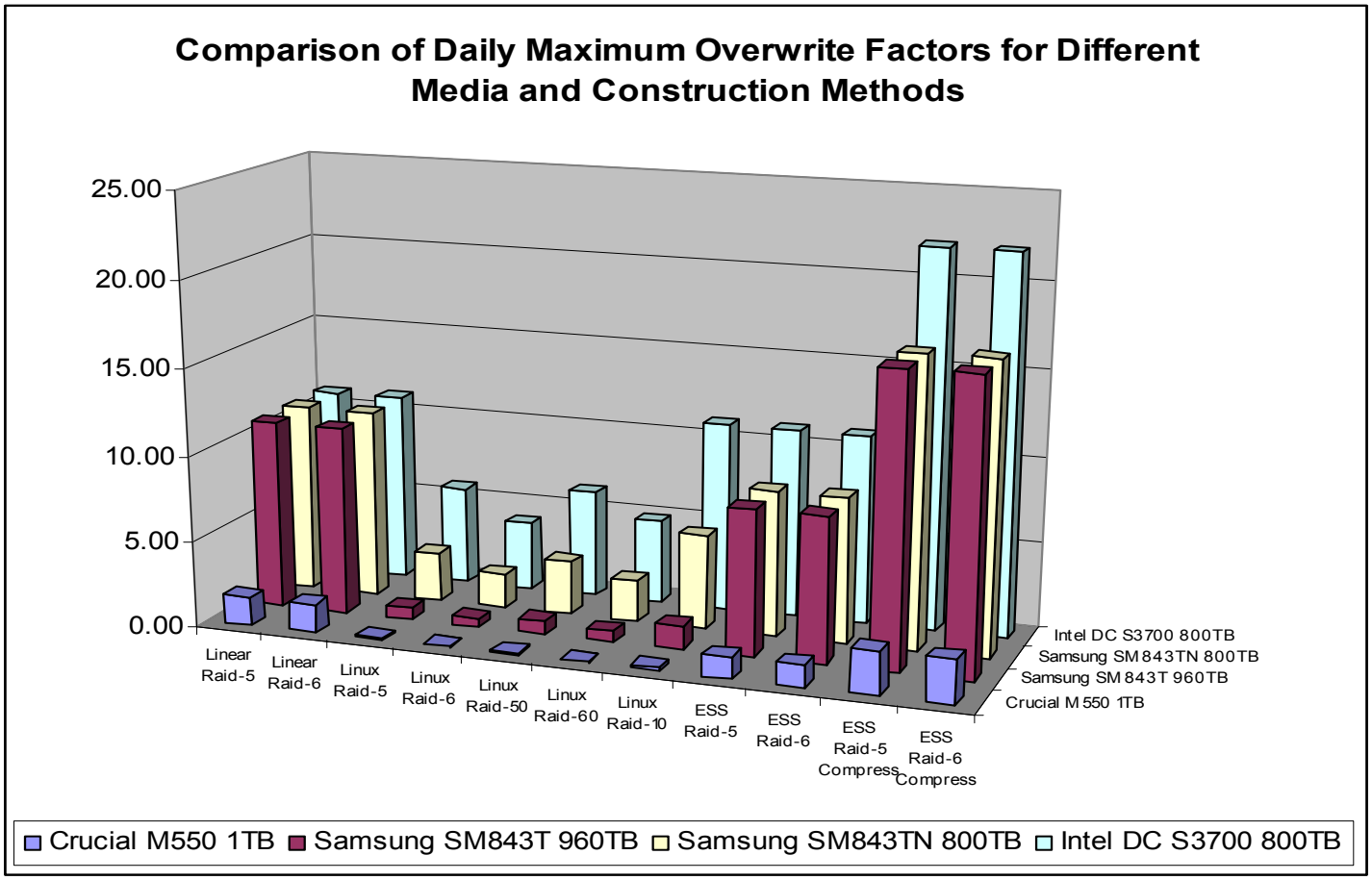
Comparison of Daily Maximum Overwrite Factors for Different Media and Construction Methods – 24 SSDs											
	Linear Raid-5	Linear Raid-6	Linux Raid-5	Linux Raid-6	Linux Raid-50	Linux Raid-60	Linux Raid-10	ESS Raid-5	ESS Raid-6	ESS Raid-5 Compress	ESS Raid-6 Compress
Crucial M550 1TB	1.64	1.64	0.11	0.07	0.12	0.09	0.21	1.33	1.33	1.33	1.33
Samsung SM843TN 960TB	10.96	10.96	0.72	0.50	0.79	0.61	1.38	8.91	8.91	8.91	8.91
Samsung SM843T 800TB	10.96	10.96	2.87	2.00	3.14	2.44	5.50	8.91	8.91	8.91	8.91
Intel DC S3700 800TB	11.00	11.00	5.74	4.00	6.29	4.89	11.00	11.58	11.58	11.58	11.58
Table 1											

An overwrite factor is the number of times a day the user addressable surface may be overwritten with the expectation that it can still be used for five years. Overwrite factors vary from

topology to topology because they inversely integrate wear generated by FTL and Raid wear amplification, normalizing the value relative to usable space. By example, we see that the Raid-10 Overwrite Factor for a Crucial drive is about 0.21, while the Raid-5 variant is only 0.11. The variation is explained by the fact that Raid-5 needs only 1 parity drive for the 24 SSDs in this study, while Raid-10's redundancy spreads over 12 SSDs, which decreases the addressable surface by almost 2:1, increasing the permissible overwrites as seen from the perspective of net space.

For Comparison purposes, this table, from left to right, looks initially at linear durability, then at random durability. Next it looks at Raid 50/60 combinations designed to increase random write speeds. Then it looks at the fastest, most durable generic Linux solution: Raid-10. Finally, this is followed up by ESS methodologies, and some approximation of the impact of compression and deduplication upon ESS overwrite factors.

Below are the graphic comparisons of these results. In considering these, you need to recognize that there is a 300-fold difference between the durability extremes (Crucial Raid-6 and ESS compressed Intel) represented.



-- Chart 1 –

High Speed mass storage can be broken down into three grades of durability: low, medium or moderate, and heavy duty.

Low duty applications generally update far less than 5% of the addressable landing surface on a daily basis. Similarly, light duty applications are often write-once read-many extremely large files such as gigabyte or even terabyte high resolution video clips, which inherently (when properly managed) result in high levels of linear writing on Raid-stripe boundaries. In such cases, as Table 1 infers, the least durable flash media can be used, and ESS is not necessary or useful absent the usefulness of a special property of ESS such as high speed block compression.

Medium duty applications tend to generate 10% to 50% of a full overwrite of addressable media per day of use. For instance, an email server providing both IMAP and POP3 service is likely to generate 20% to 30% overwrites on an average day.

Using this rule of thumb and Table 1, it is clear that Commercial grade SSDs such as the Crucial M550 referenced above are not suitable for moderate duty Enterprise environments when used with ordinary Linux Raid-5, -6, or -10 because of the problems of wear- and Raid-amplificiation. However, ESS with its linearization clearly is suitable. For instance, it increases safe overwrite levels of a Raid-6 topology from 0.07 to 1.26, an eighteen-fold increase.

Drawing from the data on Table 1, it is clear that any moderate duty "Enterprise" drive such as the Samsung SM843TN will work well with medium duty applications when managed by either a Raid-10 environment or ESS environment. There may be questions about durability in parity environments such as Linux Raid-5 or -6. However, because of the slowness of parity-based writing (see discussion below) absolute possible daily write levels in such Enterprise drives are generally below the "safe" levels in Table 1.

When applications are clearly heavy duty, the use of high end media with 20,000 or more erase cycles and low wear amplification such as the Samsung SM843T or the Intel DC S3700 is desirable. Such heavy duty products will assure a robust safety margin, as will lesser drives when used with ESS.

The random write performance problem of Flash memory. As indicated earlier, Flash memory once had its own random write problem, which has largely been addressed. Random write performance in Flash media now has peak performance approaching 50,000 IOPS per SSD, while PCIe devices often more than treble that rate. But this leaves another problem: the Raid random write problem, which varies based upon the Raid type.

As previously mentioned, Raid-10 always has to write both its base data and then mirror a copy to a second set of drives. Accordingly, on a 24 drive set, only 12 drives worth of writing can concurrently occur. By comparison, a Raid-0 set of 24 drives will accept data twice as fast as the Raid-10 set.

But the random write problem of Raid-10 is not as severe as that of Raid-5 or Raid-6. In the case of Raid-5, all drives except the data target and the parity stripe must be read. Then the parity must be calculated, and the other two drives must be written to. Raid-6 is similar: all the data drives except the target must be read, two parities calculated, and three drives written.

In this situation, the congestion of all the drives in a Raid-5 or Raid-6 set means that a device can only be written to at the random write speed of a single device: a twelfth of the Raid-10 24 drive example above. There are few design ways through this problem. Hardware solutions such as Adaptec's newest smart Raid card with their alternate parity mechanism have low limits to both read and write performance. Accordingly, while these may marginally improve the random write performance of slow Raid-5 systems, they tend to massively slow down the random read speed due to computational time losses. To give this some scope, the newest HBAs random read at over a million IOs a second, while even the prior generation could read close to a half-million IOs a second. Conversely, smart controllers have grave difficulties reaching even 100,000 read IOPS even though smart Raid controllers typically are two to four times more expensive than HBAs.

About the only traditional approach that does work to improve random write speed is to split larger devices into smaller arrays and couple the small arrays in so-called Raid-50 or Raid-60 structures. Three stripes so managed will random write three times faster than single Raid-5 or Raid-6 stripes. But on large storage sets, even a 100% or 200% improvement of Raid-5 or -6 performance will still be a small portion of Raid-10 performance. And there is a significant media cost penalty in using Raid-50 -60 configurations. In general, they cost more per usable terabyte than do ESS solutions, while delivering a small fraction of the durability and performance of ESS solutions.

How ESS solves the random write performance problem of Flash memory. ESS writes atomic clusters of random writes, FIFO, complete with metadata, as linear writes, with segments spanning entire Raid stripes and linearly writing entire erase blocks. As a result, ESS writes at the composite linear speed of all the SSDs present, and also avoids the need to do any random reading to calculate Raid-5 or Raid-6 parities.

This write mechanism is very, very fast. Our latest test, using older 2012 SSD technology, resulted in more than 1.6 million random 4KB write IOPS with 24 SSDs. (See the end of this paper for a detailed set of performance tests on one set of equipment with commentary.) With current generation SSDs, bus speeds, and computational resources, performance will be higher. But bleeding edge testing is not very useful when ESS is already five to a hundred times faster than ordinary solutions. Table 2, below, will look at the issues of relative random write performance.

A summary of the random write performance advantage of ESS. In the following table, we summarize the random write performance of the various topologies for each of the sampled drives, including both the linear-engine-only and compression/deduplication versions of ESS. These figures, which are easily calculable but which are confirmed in principal by repeated testing, confirm the conclusions previously made.

4KB Random Write Performance of Various Topologies and SSDs in IOPS – 24 SSDs											
	Linear Raid-5	Linear Raid-6	Linux Raid-5	Linux Raid-6	Linux Raid-50	Linux Raid-60	Linux Raid-10	ESS Raid-5	ESS Raid-6	ESS Raid-5 Compress	ESS Raid-6 Compress
Crucial M550 1TB	n/a	n/a	80,000	80,000	240,000	240,000	960,000	1,531,800	1,452,000	966,000	924,000
Samsung SM843TN 960TB	n/a	n/a	15,000	15,000	45,000	45,000	180,000	1,531,800	1,452,000	966,000	924,000
Samsung SM843T 800TB	n/a	n/a	15,000	15,000	45,000	45,000	180,000	1,531,800	1,452,000	966,000	924,000
Intel DC S3700 800TB	n/a	n/a	37,000	37,000	111,000	111,000	444,000	1,531,800	1,452,000	966,000	924,000
Table 2											

How ESS drives down high speed storage costs. Enterprise Storage Stack reduces the cost per user addressable unit of storage in three fundamental ways.

First, ESS allows you to build high speed, high durability storage using Raid-5 or Raid-6 parity striping mechanisms rather than using Raid-10 mirroring. In the ESS schema, Raid-5 actually random writes faster than Raid-10, because there are more user addressable landing surfaces. For the same reason, when managed by ESS, Raid-5 actually delivers more practical durability per unit of gross storage than does Raid-10. This ability effectively halves the materials cost of building a user addressable unit of high performance mass storage.

As can be seen in Table 3, an ESS-configured Raid-6 storage set typically costs about 40% less per user addressable terabyte than does the same drive set configured Raid-10. Similarly, in most cases, ESS costs only 10% more than its Linux Raid-5 or Raid-6 country cousin even though it typically will run 30 times faster (see Table 2), and be three to four times as durable (see Table 1).

Cost per Gigabyte of Various Topologies and SSDs											
	Linear Raid-5	Linear Raid-6	Linux Raid-5	Linux Raid-6	Linux Raid-50	Linux Raid-60	Linux Raid-10	ESS Raid-5	ESS Raid-6	ESS Raid-5 Compress	ESS Raid-6 Compress
Crucial M550 1TB	0.61	0.63	0.61	0.63	0.66	0.77	1.16	0.83	0.86	0.52	0.53
Samsung SM843TN 960TB	1.28	1.34	1.28	1.34	1.40	1.64	2.45	1.55	1.61	0.87	0.90
Samsung SM843T 800TB	1.54	1.61	1.54	1.61	1.68	1.96	2.95	1.82	1.89	1.01	1.05
Intel DC S3700 800TB	2.78	2.90	2.78	2.90	3.04	3.55	5.33	3.13	3.26	1.67	1.73

Table 3

Second, if you desire, ESS allows you the ability to substitute less expensive Flash media in lieu of a higher durability product without sacrificing practical durability or performance. For instance, a Crucial M550 Commercial SSD managed with ESS has almost exactly the same durability as the Samsung Enterprise SM843TN drive configured Raid-10, while offering more random write performance. Similarly, the Samsung SM843TN moderate duty Enterprise drive, when managed by ESS, delivers more performance and durability than the high durability SM843T while just about matching the durability of the Intel DC S3700 and exceeding its performance. Substitutions of this sort can singularly reduce manufacturing costs per addressable unit of space by a further 20% to 50% without reducing storage performance or durability.

Finally, ESS incorporates high speed, real time compression and deduplication as software options. Historically, it has been common to think of dedupe and compression as features rather than quantities for several reasons. The first was that many implementations were so resource-intensive that they needed to be processed and applied after the fact, rather than in real time. The second was that many of the solutions were applied at the file management level rather than at the block layer. The final reason is that the compressibility of data varies widely among data sets, and it is only when storage units become many terabytes in size that averages begin to have meaning.

While there are some elements which will not compress at all (examples: gifs, jpegs, and previously encrypted data) many systems will compress and/or dedupe 2x or more, and some environments such as backup, VDI, and virtual systems may compress 5-fold to 20-fold through deduplication alone. In the cost reduction examples above, we have assumed an average compression of just 2x.

In considering all of the above, as well as performance and durability tradeoffs, it is important to remember that the factors talked about are multiplicative. Using ESS and Raid-5 or -6 will reduce the cost per terabyte by 40% relative to Raid-10. Substition will reduce the remaining 60% by 20% to 50%, and compression techniques will close to halve again what remains.

Conclusions: while ESS does not normally create benefits for users of light-duty write structures, system developers providing moderate duty or heavy duty solutions can offer their customers and selves the best of both worlds, on the one hand improving performance and durability while on the reducing build costs and thus increasing competitiveness and/or profitability.

Relative Performance Tests

On 21 July 2014, EasyCo conducted a series of random write performance tests against a chassis built with 1 SuperMicro X9-SRL-F single socket Motherboard, 1 Intel E5-1650 six core 3.2ghz with 64GB of RAM, 3 LSI 9207-8i HBI disk controllers, and 24 SSDs consisting of 16 Samsung 830 128GB SSDs, and 8 SanDisk Ultra Plus 128GB SSDs.

The testing software in all cases was EasyCo's bm-flash performance benchmark. The test first tested a number of ESS configurations, including linear, compressed, and compress/dedupe configurations at different data sizes. These were followed up by testing standard Linux Raid configurations in like manner. Once the IOPS rates were determined, performance in megabytes per second was computed by multiplying the IOPS rate by the relevant size. The following are the results.

IOPS Performance For Various Read/Write Sizes								
	ESS Raid-5 Random Write					Linux Native Software Random Write		
Block Size	ESS Linear	Compress 0%	Compress 75%	Dedupe Compress 0%	Dedupe Compress 75%	Raid-0	Raid-5	Raid-10
4K	1,539,153	919,104	1,235,525	676,851	858,915	348,876	45,752	168,444
8K	793,145	483,171	833,835	383,991	415,304	380,053	27,853	166,757
16K	385,047	269,168	467,466	211,352	229,983	245,614	16,061	142,974
32K	202,547	138,838	249,584	107,641	115,172	146,975	8,405	87,062
64K	103,331	71,940	129,918	55,567	60,031	81,577	4,197	46,649
128K	51,577	36,198	66,701	32,207	30,933	38,862	2,140	16,598
256K	25,705	19,462	34,574	16,395	15,604	23,052	1,096	9,149
Megabytes per Second Performance For Various Read/Write Sizes								
	ESS Raid-5 Random Write					Linux Native Software Random Write		
Block Size	ESS Linear	Compress 0%	Compress 75%	Dedupe Compress 0%	Dedupe Compress 75%	Raid-0	Raid-5	Raid-10
4K	6,012	3,590	4,826	2,643	3,355	1,362	178	657
8K	6,196	3,774	6,514	2,999	3,244	2,969	217	1,302
16K	6,016	4,205	7,304	3,302	3,593	3,837	250	2,233
32K	6,329	4,338	7,799	3,363	3,599	4,592	262	2,720
64K	6,458	4,496	8,119	3,472	3,751	5,098	262	2,915
128K	6,447	4,524	8,337	4,025	3,866	4,857	267	2,074
256K	6,426	4,865	8,643	4,098	3,901	5,763	274	2,287

Table 8

What is interesting about these results is that while the ESS as-built numbers roughly correspond to the theoretical performance of the device, the random performance in a Linux environment is far more dependent upon the number of write threads.

ESS coalesces writes into a single write structure that is processed by a multi-stage pipeline running across several cores. This structure accelerates low queue depth writes by creating an "apparent" write latency of < 1uS or more than 50x faster than a traditional SSD. In that many applications only write from a single thread (this is true for most filesystems), the ESS write behavior for low thread counts can result in huge performance gains beyond what is available when driving large numbers of clients. Conversely, the Linux writes here are based upon 100 concurrent write threads, which is a great deal of activity. When write threads decrease, performance is significantly impacted.

Source Data Tables and Explanatory Notes

The data and conclusions referenced above are based primarily on the following four tables, Table 4 through Table 7. Each individual table analyzes one brand-and-model of Commercial grade or Enterprise grade SSD. Analysis is performed on two axes.

The left to right axis covers various Raid-structures. These begin with Raid-5 and -6 cost and performance when built for use as an engine receiving exceptionally large files which will largely be linearly written rather than randomly written. The next two cover random writing of data in Raid-5 and -6 sets. The third pair cover Raid-50 and -60 combinations to analyze the impact of using this mechanism to speed up random writing. The fifth cluster covers systems built with traditional Raid-10 methodologies. The last two pairs cover systems managed by ESS. The first pair covers use of basic ESS together with a Raid-5 or -6 configurations. The second pair covers the same configurations, but with our data compression methodologies active.

The vertical elements are broken into three groups.

The first (Yellow) group of two reports both the gross amount of storage space (which is the same for all Raid options) and the net amount of directly user addressable space made available by each of the options. In determining net space for ESS compressible configurations, we arbitrarily assume that compression will equal 50% of all data storage.

The second (Blue) group of five covers various costs of manufacturing storage appliances, and also computes the cost per user addressable terabyte. The costs here are broken down into several categories. The chassis cost presumes the use of a 2u 24 bay hot swap case, three LSI 9300 series HBAs, 64GB of RAM, and a 6 core high speed Intel CPU. The media cost is based upon prices obtained either at Newegg.com or Google Shopping. The ESS license cost is based upon the ESS entry level license fee. License fees for more committed licensees can decline as much as 75%. The cost per terabyte is the total system cost divided by the landable surface: the user addressable space.

The third (Purple) group of two covers both estimates of and tested results of random write speeds as well as the allowable overwrite levels at each topology point.

Linux random write performance is a multiplicand of the reported random write speeds for each drive adjusted by the drive limits which Linux Raid places upon these topologies. (Based upon the testing reported above, this appears to be an overly-generous assumption.) ESS writes are based upon tested performance of 24 drive sets of the basic engine and the advanced engine, adjusted for the number of Raid stripes present.

Permissible overwrites per day are computed by either using net published daily durability (as in the Intel S3700) or taking lifetime erase cycles times gross storage space, and then dividing the same first by the number of days in five years, then by the wear amplification, and next by the raid amplification, before finally dividing the product by net user addressable storage.

The Crucial M550 is a Commercial grade SSD built with 3,000 erase cycle Flash, which appears to have an 8:1 internal wear amplification, based upon its warranty.

The Samsung SM843TN and SM843T are both classed as Enterprise SSDs. Both are made with 20,000 erase cycle Flash. The TN variant has limited free space and accordingly has internal wear amplification of 8:1. The T variant is made with approximately 20% more free space and typically has 2:1 internal wear amplification.

The Intel DC S3700 is a heavy duty Enterprise drive warranted as supporting 11 overwrites per day without specification as to Flash type or durability, or internal wear amplification.

24-Drive Array of Crucial M550 1TB Commercial Grade SSD											
	Linux Software Raid							Enterprise Storage Stack - Random IO			
	Linear IO		Random IO					Raid-5	Raid-6	Raid-5 Compress	Raid-6 Compress
	Raid-5	Raid-6	Raid-5	Raid-6	Raid-50	Raid-60	Raid-10				
Gross Surface	24,576	24,576	24,576	24,576	24,576	24,576	24,576	24,576	24,576	24,576	24,576
Landable surface	23,552	22,528	23,552	22,528	21,504	18,432	12,288	22,374	21,402	44,749	42,803
Chassis Cost	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600
Media Cost: 24x1024GB SSDs	10,680	10,680	10,680	10,680	10,680	10,680	10,680	10,680	10,680	10,680	10,680
ESS License Fees	n/a	n/a	n/a	n/a	n/a	n/a	n/a	4,400	4,200	8,800	8,400
Total Build Cost	14,280	14,280	14,280	14,280	14,280	14,280	14,280	18,680	18,480	23,080	22,680
Cost per addressable gigabyte	0.61	0.63	0.61	0.63	0.66	0.77	1.16	0.83	0.86	0.52	0.53
4KB Random Write Ops/sec	n/a	n/a	80,000	80,000	240,000	240,000	960,000	1,531,800	1,452,000	966,000	924,000
Overwrites per day	1.64	1.64	0.11	0.07	0.12	0.09	0.21	1.33	1.33	1.33	1.33

Table 4

24-Drive Array of Samsung SM843TN 960GB Limited Freespace Enterprise SSD											
	Linux Software Raid							Enterprise Storage Stack - Random IO			
	Linear IO		Random IO					Raid-5	Raid-6	Raid-5 Compress	Raid-6 Compress
	Raid-5	Raid-6	Raid-5	Raid-6	Raid-50	Raid-60	Raid-10				
Gross Surface	23,040	23,040	23,040	23,040	23,040	23,040	23,040	23,040	23,040	23,040	23,040
Landable surface	22,080	21,120	22,080	21,120	20,160	17,280	11,520	20,976	20,064	41,952	40,128
Chassis Cost	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600
Media Cost: 24x960GB SSDs	24,672	24,672	24,672	24,672	24,672	24,672	24,672	24,672	24,672	24,672	24,672
ESS License Fees	n/a	n/a	n/a	n/a	n/a	n/a	n/a	4,200	4,000	8,400	8,000
Total Build Cost	28,272	28,272	28,272	28,272	28,272	28,272	28,272	32,472	32,272	36,672	36,272
Cost per addressable gigabyte	1.28	1.34	1.28	1.34	1.40	1.64	2.45	1.55	1.61	0.87	0.90
4KB Random Write Ops/sec	n/a	n/a	15,000	15,000	45,000	45,000	180,000	1,531,800	1,452,000	966,000	924,000
Overwrites per day	10.96	10.96	0.72	0.50	0.79	0.61	1.38	8.91	8.91	8.91	8.91

Table 5

24-Drive Array of Samsung SM843T 800GB Heavy Duty Enterprise SSD											
	Linux Software Raid							Enterprise Storage Stack - Random IO			
	Linear IO		Random IO					Raid-5	Raid-6	Raid-5 Compress	Raid-6 Compress
	Raid-5	Raid-6	Raid-5	Raid-6	Raid-50	Raid-60	Raid-10				
Gross Surface	19,200	19,200	19,200	19,200	19,200	19,200	19,200	19,200	19,200	19,200	19,200
Landable surface	18,400	17,600	18,400	17,600	16,800	14,400	9,600	17,480	16,720	34,960	33,440
Chassis Cost	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600
Media Cost: 24x800GB SSDs	24,672	24,672	24,672	24,672	24,672	24,672	24,672	24,672	24,672	24,672	24,672
ESS License Fees	n/a	n/a	n/a	n/a	n/a	n/a	n/a	3,600	3,400	7,200	6,800
Total Build Cost	28,272	28,272	28,272	28,272	28,272	28,272	28,272	31,872	31,672	35,472	35,072
Cost per addressable gigabyte	1.54	1.61	1.54	1.61	1.68	1.96	2.95	1.82	1.89	1.01	1.05
4KB Random Write Ops/sec	n/a	n/a	15,000	15,000	45,000	45,000	180,000	1,531,800	1,452,000	966,000	924,000
Overwrites per day	10.96	10.96	2.87	2.00	3.14	2.44	5.50	8.91	8.91	8.91	8.91

Table 6

24-Drive Array of Intel DC S3700 800GB Heavy Duty Enterprise SSD											
	Linux Software Raid							Enterprise Storage Stack - Random IO			
	Linear IO		Random IO					Raid-5	Raid-6	Raid-5 Compress	Raid-6 Compress
	Raid-5	Raid-6	Raid-5	Raid-6	Raid-50	Raid-60	Raid-10				
Gross Surface	19,200	19,200	19,200	19,200	19,200	19,200	19,200	19,200	19,200	19,200	19,200
Landable surface	18,400	17,600	18,400	17,600	16,800	14,400	9,600	17,480	16,720	34,960	33,440
Chassis Cost	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600	3,600
Media Cost: 24x800GB SSDs	47,520	47,520	47,520	47,520	47,520	47,520	47,520	47,520	47,520	47,520	47,520
ESS License Fees	n/a	n/a	n/a	n/a	n/a	n/a	n/a	3,600	3,400	7,200	6,800
Total Build Cost	51,120	51,120	51,120	51,120	51,120	51,120	51,120	54,720	54,520	58,320	57,920
Cost per addressable gigabyte	2.78	2.90	2.78	2.90	3.04	3.55	5.33	3.13	3.26	1.67	1.73
4KB Random Write Ops/sec	n/a	n/a	37,000	37,000	111,000	111,000	444,000	1,531,800	1,452,000	966,000	924,000
Overwrites per day	11.00	11.00	5.74	4.00	6.29	4.89	11.00	11.58	11.58	11.58	11.58

Table 7

Status: Draft
Prepared by: Sam Anderson, CEO, EasyCo LLC
Date: 25 July 2014

EasyCo LLC
220 Stanford Drive
Wallingford PA 19086 USA
Tel: (+1) 610-237-2000
Free: 888-473-7866
Email: sales@EasyCo.com
Web: http://WildFire-Storage.com