Exposing the Unvarnished Truth About Database Storage

Why DBAs Today Have to Be Storage Experts
Executive Summary

The world of database storage has changed and continues to change at an ever-increasing pace. It’s become virtualized, networked, distributed, and ultimately far more complicated for the database administrator (DBA) than ever before. Take the simple concept of a disk drive. DBAs historically could assume it meant a spinning hard disk drive (HDD). They could control that physical device down to where on the platter they put their data. They could stripe it, short-stroke it, even fool it into behaving just the way the DBA wanted it to.

Not today. The relentless implementation of data center virtualization has decoupled the representation of the storage device from the physical storage thus frustrating historic DBA efforts. A drive today can still mean a HDD. But it is far more likely to mean a virtualized storage unit that’s part of physical HDDs, RAID, virtual RAID, multiple concurrent RAID arrays, NAS, multiple concurrent NAS systems, even solid-state storage (a.k.a. solid state drives or SSDs). All of these technologies are supposed to simplify storage for the DBA. This is how they are marketed and sold. Candidly, they can simplify the DBA’s life somewhat when it comes to implementation. Operations, management, performance tuning, storage consumption, and storage troubleshooting are a completely different story. All of these tasks have become more, not less, manual and complicated.

This document scrutinizes how the decoupling of the storage representation from the physical storage, otherwise known as virtualization, has made the life of the DBA a living storage hell by making their jobs more not less difficult. It then examines the storage problems and storage obstacles that the DBA has to deal with today including ones they never had to deal with before.

It additionally details how common workarounds to these database storage problems are principally a brute force approach that throws hardware at these problems deferring versus solving the problems. Then it critically assesses how well Oracle’s latest ZS3 application engineered storage (Oracle’s co-engineered combination of the database, OS, file system, and storage) solves these database storage problems.
# Table of Contents

**Executive Summary** .................................................................................................................. 2

**Database Storage Problems** ........................................................................................................ 4

- Database storage performance tuning ........................................................................ 4
  - Typical database storage performance tuning “workaround” ........................................ 5
- Slowing/reducing database storage capacity consumption ........................................ 6
  - Common database storage consumption “workarounds” ............................................. 6
- Troubleshooting database storage problems and performance bottlenecks .................. 6
  - Typical database storage troubleshooting “workarounds” ............................................. 6
- Accelerating database backups to stay within backup windows .................................. 7
  - Mainstream database backup window “workarounds” ................................................ 7
- Database storage active archiving .............................................................................. 8
  - Frequent database storage active archiving “workarounds” ........................................ 8
- Managing database storage with fewer manually labor-intensive tasks ................... 8
  - Regular database storage manually labor-intensive task “workarounds” ....................... 8
- Finding a way to reduce the total cost of ownership (TCO) ........................................ 9
  - TCO reduction “workarounds” ................................................................................... 9

**Oracle’s ZS3 Next Generation of Application Engineered Storage (AES)** ......................... 9

**How the ZS3 Tackles Database Storage Problems** ................................................................. 9

- ZS3 tuning of the storage hardware to get maximum database performance results .... 9
  - 3rd Gen Hybrid Storage Pools (1st Gen in 2008) ......................................................... 10
  - Dynamic SSD Optimization (DSO) .............................................................................. 10
  - In-Memory Deduplication ......................................................................................... 10
  - Parallel Access Sequencing ..................................................................................... 10
  - Infiniband Access .................................................................................................. 10
- ZS3 slowing/reducing database storage capacity consumption .................................. 11
  - HCC ....................................................................................................................... 11
  - HCC with Advanced Data Optimization ................................................................. 11
- ZS3 troubleshooting database storage problems and performance bottlenecks .......... 11
  - ZS3 DTrace Analytics .......................................................................................... 11
- ZS3 accelerating database backups to stay within backup windows ......................... 12
  - ZS3 Oracle Database data protection engineering ................................................. 12
- ZS3 database storage active archiving ...................................................................... 12
  - Database partitioning and heat maps ..................................................................... 12
  - Flashback Data Archiving (FDA) .......................................................................... 13
- ZS3 management of labor-intensive tasks .................................................................. 13
  - OISP ...................................................................................................................... 13
- ZS3’s ability to reduce TCO for database storage ....................................................... 13

**Summary and Conclusion** ..................................................................................................... 14
Database Storage Problems

Decoupling the storage image from the physical storage (virtualization) has solved several serious storage problems. It allows storage to be thin or fat provisioned, scaled, protected, deduplicated, compressed, and replicated online without downtime. These are all good things as far as IT is concerned because it dramatically slashes scheduled application downtime. But what it does to the DBA is something different. It now requires the DBA to comprehensively fathom the storage systems they’re utilizing for their database systems. DBAs need to understand how the virtualization of that storage affects their ability to meet or exceed their database needs and requirements.

Storage virtualization can occur in the storage system or a server virtualization hypervisor. It is the basis behind software-defined storage (SDS). Storage virtualization hides a lot of information from the DBA. A virtual LUN or virtual disk is not the same as a physical LUN or disk. It can be spread over multiple physical LUNs, multiple/different RAID groups/sets, different drive performance, different storage tiers, and even different storage systems. In fact, it’s possible that there can be and often are, multiple layers of storage virtualization between the database and the actual physical storage. These multiple software layers add latency to database transactions. Transactions do not like latency. Worse than latency is the blind sharing of storage assets. Virtualization is designed to increase hardware utilization and it does a very fine job of doing just that. But it requires multiple applications and databases overlapping on the actual physical storage to get those higher levels of utilization. For the DBA this is invisible because they do not know how the physical storage and infrastructure is actually constructed. They will only see the symptoms, which are inconsistent latencies and response times. The storage administrator also might or might not truly know, depending on their level of expertise. This means the DBA must either trust the storage administrator relying on their skills at making sure the DBA’s database requirements are fully understood and met; or the DBA must develop comprehensive knowledge of the database storage. Either way, there are minimally seven different storage problems that DBAs have to cope with today. They include:

- Tuning the storage hardware to get maximum database performance results;
- Slowing/reducing database storage capacity consumption;
- Troubleshooting database storage problems and root causes of performance bottlenecks;
- Accelerating database backups to stay within backup windows;
- Database storage active archiving;
- Managing with fewer manually labor-intensive tasks;
- Finding a way to reduce the total cost of ownership (TCO).

Database storage performance tuning

Databases have consistently placed extraordinarily high performance demands on storage. This is especially true for OLTP and OLAP. DBAs have always strived for highly consistent fastest possible IO. It’s IO performance consistency or predictability that’s valued even more highly than IO peak speed. Delivering fast consistent IO performance is non-trivial when considering database IO high randomization, the fact that data in the database keeps growing exponentially, or the inconsistency of TCP/IP networking.

Mastering this problem used to mean the DBA placed data only on the outer edge of the HDDs, a.k.a. short stroking. It also meant making sure the number of drives in a RAID group were balanced, and the speed of the drives were preferably 15,000 RPM FC or SAS drives. Managing database storage performance in this manner required knowledge, skill, and most importantly, experience. Effective management was as much feel, as it was science. This approach wasted a lot of HDD capacity and was not for the faint-of-heart. It also no longer makes sense today as databases continue to grow larger, scaling to tens or hundreds of TBs and even PBs on the horizon.

Virtualizing the storage changed the game. Effective database storage performance today requires a lot more storage system knowledge and control. Virtualization masks how the storage system or hypervisor is actually managing the underlying
physical storage media. Volumes or data stores are commonly shared by multiple applications, unbeknownst to the applications, users, or DBAs. Tools that provide the insight and control are available; however, there are unintended consequences. Most storage systems are optimized to provide the greatest good for the greatest number of users and applications. Manipulating the underlying rules can have horrendous negative impact on overall storage performance, capacity utilization, data protection, and application performance. The vendor’s warranty can potentially be voided—which is why most DBAs take another path to managing database storage performance tuning.

**Typical database storage performance tuning “workaround”**

The path of least resistance is to implement and utilize Flash SSDs, within the database server(s) and/or within the database storage system, to deal with troublesome database storage performance issues. The conventional wisdom is that Flash SSDs provide very high IOPS, are easy, simple, reliable, convenient, and require minimal skills on the part of the DBA. This idea is based on the concept of throwing storage IOPS at the problem until it goes away. As with the majority of conventional wisdom, there are issues and flaws glossed over in this simplistic view.

DBAs need to understand the issues with Flash SSDs to effectively utilize them. It starts with the basic technology. Whereas HDDs are a bit storage media, Flash SSDs are a block storage media. Data written to HDDs can be modified, added to, or deleted (pointers erased) without any degradation on read or write performance. Data written to a page on a SSD is a different can of worms. Written data can’t be modified in any way. Files or data that change are written to a new page. The old data page is designated for erasure. Designated erasure pages are collected during garbage collection, erased, and then designated as available for new data. Flash has a limited number of write/erase cycles also know as its wear-life. That wear-life is affected by the size of the pages. Pages can be small such as 4KB sizes or large such as 64KB. Smaller pages are more efficient for small data sizes typically leaving a lot less of the page unused and a lot more pages available to be written to. Small pages reduce what’s known as write amplification but with a cost. There are many more pages (orders of magnitude more) for the SSD controller to track for the garbage collection consuming cycles and adding latency. Larger page sizes have lower latencies (fewer pages to track) but far more waste, especially in a database’s random IO environment. This increases the write amplification while reducing the wear-life. The DBA must be aware of the wear-life to make effective use of the Flash SSD and prevent data loss from SSD Flash (NAND) failures. Using Flash SSDs as a read cache (a.k.a. write-through cache) will produce much less wear than using them for write-back cache because of the much greater number of writes. Sound complicated and non-intuitive? It is. And that’s only the beginning. Tiering and caching have different negative affects on both the Flash SSD’s performance and wear-life.

Flash SSD performance will also degrade over time and level off;—it will never be as fast as when it was first utilized. Flash SSD’s best performance is the first write out of the box. Then there is the storage processor (CPU) bottleneck. SSD IOPS often overwhelm the system CPU (in the storage system or the application server,) shifting the IO bottleneck from the storage media to the processor.

All of this does not mean that Flash SSDs are bad. It means that they are different. The DBA must learn to leverage this technology, unlearning what they used to do with HDDs. All of the issues can be dealt with via expertise, experience, and feel. Some storage vendors have put a layer of software and virtualization above the SSDs that hide the Flash SSDs and these issues from the DBA. It also makes performance tuning, troubleshooting, hot spot analysis, and congestion determination quite difficult and time consuming.

There are several things to remember about this workaround. Throwing SSD IOPS at the problem is a brute force approach and an expensive one at that. SSDs cost quite a bit more than HDDs of equivalent capacity. The vastly increased IOPS will provide database performance relief just about immediately. Ultimately, this approach is a treatment of the symptoms, not the root cause of database storage performance problems. Every storage system whether it is external or internal, will have an architectural IOPS ceiling. That ceiling will be reached regardless of the number of SSDs and the SSD IOPS ratings. When that ceiling is reached, the DBA once again must intervene to solve the database storage...
performance problems. This is a direct result of the lack of database to storage intercommunications, awareness, or even integration that would enable them to work hand-in-hand.

- **Slowing/reducing database storage capacity consumption**

   It’s no secret that databases are growing rapidly doubling in size every 18 to 24 months according to IDC. That database data consumes storage. As the database data grows, so does the database storage. Why is it then that the database storage grows far more rapidly than the primary database data? It is because data begets more data. The primary database data will have multiple replicas. Replicas are created for dev-test, data mining, analysis, disaster recovery, business continuity, backup, archive, etc.

   - **Common database storage consumption “workarounds”**

     The DBA’s first thought at reducing database storage consumption is to employ data reduction technologies. The data reduction technologies commonly utilized include thin provisioning, data deduplication, and data compression.

     Thin provisioning fools the database into thinking it has more storage than it actually has. Thin provisioning allows less of the physical storage to be allocated enabling it to be shared by multiple applications. It doesn’t actually reduce storage consumption. It reduces storage waste. Unfortunately, the Oracle Database is a bit hard to spoof. The Oracle Database takes a hard look at all of the storage capacity it has been assigned as told by the storage system. If it can’t actually see all of the alleged capacity, it assumes it does not have what it cannot see, negating the value of thin provisioning.

     Data deduplication eliminates duplicate blocks and/or files in all of the data in the storage system. Data compression eliminates duplicate blocks within a file or data set as well as null values. The data reduction ratios generally range from 2x to 3x depending on the datatype. But the downside of these deduplication and compression ratios is increased latency on every transaction, which equates to reduced database performance. Writes must wait for the deduplication or compression algorithm to complete. Reads must wait for the rehydration of the data to complete. There is no database to storage intercommunications, awareness, or integration that could possibly allow the database and or the storage to adjust, manage, and potentially correct the situation.

- **Troubleshooting database storage problems and performance bottlenecks**

   As previously discussed, the latest generation of virtualization obfuscates the visibility between the database and the database storage. Thus making it quite difficult to troubleshoot database storage problems. Finding the hot spots and performance bottlenecks is a painstaking, time consuming, manually labor-intensive process. It requires trapping all of the data end-to-end when the problem occurs. And it usually requires trapping it more than once. It’s a patience trying exercise.

   - **Typical database storage troubleshooting “workarounds”**

     Trapping the problem requires third-party performance management software. A client piece of software (a.k.a. an agent) is installed on the database server. The performance management server communicates with the storage as well as the storage network switch through their respective APIs. Then it correlates the database performance variances with the information collected from the storage system and the network. This type of product doesn’t necessarily pinpoint the root cause of problems and performance, but does tend to indicate where to look.

     Like all workarounds, performance management software has its downside. The agent consumes server cycles normally utilized by the database software. It is yet another software package to license, (more OpEx costs), manage, learn, operate, maintain, and update in addition to the database, server, OS, hypervisor, storage system, storage network, etc.
Accelerating database backups to stay within backup windows

As databases have become bigger, their backups have become a very big problem for DBAs. This problem is known as a “blivet” or ten pounds of stuff in a five-pound rated box. There is too much data to backup within the backup window timeframe. Backups are running into the production cycle, not getting done, or getting done far less frequently (much longer recovery point objectives than required.) It’s a frustrating deteriorating situation for many DBAs.

Mainstream database backup window “workarounds”

Most DBAs try to solve the backup window problem by leveraging storage snapshots or storage mirroring. Snapshots appear to be the easy answer because they are essentially instantaneous, take nominal amounts of capacity, are replicable, writeable, easy to mount and restore from, and are simple. Appearances can be deceiving. Most snapshots are either redirect on write (ROW) or copy on write (COW). ROW has become more popular because it requires no capacity reservation for the snaps allowing more of them. Both ROW and COW snapshots are not actual copies of the data. They are copies of the directories or pointers of the data at a specific point in time. If the actual data disappears, gets lost, or becomes corrupted, the snapshots are no longer viable. These types of snapshots are good for many things; disaster recovery is not one of them. Replication of these types of snapshots can be problematic as well in that the replication is of the pointers not data.

This is why there are snapshot clones that are actual copies of the data. Clones are often replicated to another storage system on campus or in a different geographically located data center. Clones take a long time to copy; backup window issues once again come into play. Most storage system snapshots lack database awareness. The storage system does not tell the database to quiesce the database, flush the cache, and complete all writes in the proper order before a snapshot is taken. What it means is that snapshots by themselves are rarely crash consistent and can be corrupted. Several backup software products have integrated with some storage systems utilizing their agents to communicate with the database to quiesce it, flush the cache, complete the writes, and then tell the storage system to take the snapshot. These combinations require additional software licensing, agents, management, server hardware and training. Backups and especially recoveries are multi-vendor adding complexity and higher probabilities of failure. In the end, it only covers part of the problem. To do disaster recovery as well requires storage mirroring.

Storage mirroring is either synchronous or asynchronous. Synchronous mirroring makes every write twice to two different storage systems. The local write must wait for the remote write to be acknowledged before acknowledging the local write. This ensures that the remote storage system is always in the same condition as the local. From a database perspective the remote copy is always in sync but limits the distance between the two storage systems to no more than approximately 100 miles, otherwise remote acknowledgements will take longer than the application expects. The implication is that the remote storage system is in the same geographic region as the local making it more vulnerable to the same potential regional disasters such as floods, wild fires, tornados, hurricanes, etc. Asynchronous mirroring extends that distance to thousands of miles by permitting the local writes to be acknowledged before the remotes allowing a significant number of remote acks to fall behind. The result is a higher probability of a corrupted database in the event of a disaster.

Snapshots and mirroring are very costly solutions in that they require more storage systems, third-party software, high bandwidth networks, and lots of capacity for duplicate copies of the data.
The third workaround utilized by DBAs for this backup window issue and an attempt to reduce the additional capacity costs, is to use target dedupe storage appliances. Target dedupe storage appliances still require some form of backup software and/or agent. None of the appliances have done well at reducing the backup window problem, even the ones that utilize a source dedupe agent on the database server (Symantec OST or EMC DD Boost). Once again, the workarounds are dealing with the symptoms instead of the database root cause of the problem. There is little to no database and storage awareness beyond telling the database what to stop running at different points in time. These workarounds can work for a period of time with constant DBA diligence, at a very high cost, and with a good chance of failure.

Database storage active archiving

As databases grow too large they become unwieldy and response time declines dramatically. DBA common practice is to purge older passive data from the database and to archive it. Unfortunately, when that data is purged and archived, it is no longer in the database nor accessible. The data must be recovered and re-entered into the database. Doing so is time consuming, manually labor-intensive, never convenient, and far too often required.

- Frequent database storage active archiving “workarounds”

Third-party backup software and archive software are the most common workarounds. Both are incomplete solutions. Backup is never a good archive product because all of the data must be recovered in the database itself to bring back any of the old data. It’s very labor intensive and highly time consuming.

Archiving software is more popular because it creates a mountable searchable image of the data allowing only the data, transactions, rows, columns, or tables to be recovered. It’s not as painful as getting data from the backups, but it is still absolutely no fun.

Managing database storage with fewer manually labor-intensive tasks

As seen in the issues already discussed, managing database storage is a task-heavy, time-consuming effort. Each of the workarounds previously described only adds to that burden. NFS NAS is supposed to be one of the simplest storage systems to manage, and yet it too is task heavy. Take something as simple as logbias and record sizing. Optimization in an Oracle Database requires up to 4 shares per pool with the minimum Oracle recommendation of 2 pools per backup job. This in turn requires then 8 logbias tuning tasks as well as 8 record sizing tasks just to set up. When changes occur (and they always occur) those tasks must be repeated. That’s way too many manual tasks for a simple process.

- Regular database storage manually labor-intensive task “workarounds”

Homegrown scripts. DBA scripts are clever, unique, comprehensive, flexible, limited only by the creativity of the writer. And DBAs can and do have more than one. Scripting is second nature to most DBAs although most are well aware of homegrown scripts dark side.

The author of a homegrown script rarely if ever documents it. Another admin inheriting that script will either have to reverse engineer it to figure it out or most likely, start over and write a new script. That’s not the most egregious downside to scripts. Lack of quality assurance, testing, and maintenance is.

These processes are part of commercial software support. The vast majority of homegrown scripts never have them. DBAs rarely have time to write a script let alone spend much time testing it for bugs or flaws. The philosophy tends to be break/fix or caveat emptor. The lack of a QA process rears its ugly head whenever there is a change to the storage system like a microcode upgrade, or a change to the database. Basically, any change in the ecosystem can and often does break the script. Since there is no QA process or testing before the changes occur, there is no visibility as to how those changes will affect the scripts until after they’re implemented, a.k.a. hope it doesn’t break and pray it can be fixed quickly. In the end, homegrown scripts tend to create as many tasks or more than they save.
Finding a way to reduce the total cost of ownership (TCO)

All of the aforementioned database storage problems cause storage TCO to spiral out of control. Data growth is not slowing down and appears to be accelerating just as storage density gains have slowed, leading to the fastest growth in storage costs in years. IT organizations got used to storage density gains ameliorating the cost of rapidly rising data amounts under storage management. Those days are gone but IT budgets have not made the adjustment. This is never more obvious than in database storage.

TCO reduction “workarounds”

Most of these “workarounds” are focused again on the data reduction technologies previously discussed. There is a lot of unrealistic “voodoo” math associated with most data reduction technologies. Some vendors will talk in terms of “effective” storage based on incredible and incredulous data reduction ratios not seen in the real world. The actual reduction in physical storage required is usually less than expected. And the cost in database performance degradation often makes it a tough pill to swallow.

Add the relatively high costs of data protection to meet backup windows and it becomes apparent why DBAs are so frustrated when it comes to solving these intransigent database storage problems.

Oracle’s ZS3 Next Generation of Application Engineered Storage (AES)

Oracle views database storage differently from most storage vendors. Oracle sees the database through the eyes, life, and tasks of the DBA. This led them to engineer a symbiotic relationship between the Oracle Database and Oracle storage. It is only because Oracle has intimate knowledge of the entire stack from the database, to the Solaris OS, servers, Zettabyte file systems (ZFS), storage hardware, and storage software that they are able to tightly engineer them together.

Engineering the software and hardware together has allowed Oracle to tackle the database storage problems in an irreplaceable way. Database processes that would be better served in the storage system are taken on by the storage system automatically. Storage processes better handled in the database server platform are moved there as well. Dynamic flexibility allows for dynamic solutions.

How the ZS3 Tackles Database Storage Problems

First a quick review of the database storage problems. There are seven identified in this paper:

• Tuning the storage hardware to get maximum database performance results;
• Slowing/reducing database storage capacity consumption;
• Troubleshooting database storage problems and root causes of performance bottlenecks;
• Accelerating database backups to stay within backup windows;
• Database storage active archiving;
• Managing with fewer manually labor-intensive tasks;
• Finding a way to reduce the total cost of ownership (TCO).

Second, assessing the way that the next gen Application Engineered Storage, Oracle ZFS Storage Appliance, ZS3, solves these problems and how well it does so.

ZS3 tuning of the storage hardware to get maximum database performance results

The ZS3 storage auto-tunes the hardware for maximum database performance results. It does this by creating a synergistic communication between the Oracle Database and Oracle ZS3 storage in a combination of distinct Oracle exclusive technologies.
• **3rd Gen Hybrid Storage Pools (1st Gen in 2008)**

The ZS3 provides combinations of extensive amounts of DDR3 DRAM (up to 2TB/cluster), SSD write cache, SSD read cache, 10K RPM SAS drives, and 7.2K NL-SAS drives. Within Hybrid Storage Pools (HSP), data is automatically migrated from DRAM to Flash SSDs and then onto high performance, highly reliable SAS disks based on policies such as age or time passed since last access. The software is specifically engineered for multiple levels of Flash SSD storage and disk storage.

It is that extensive amount of DRAM that makes the difference in database performance. Hybrid Storage Pools take advantage of the up to 2 TB of DRAM and enables as much as 90% of the database IO to come from that DRAM automatically. DRAM is approximately 10 to 20 times faster than Flash SSDs.

• **Dynamic SSD Optimization (DSO)**

That large DRAM capacity also enables improved performance from the Flash SSDs. The ZS3 solves the Flash SSD write amplification performance conundrum previously discussed. To get the longest wear life out of a Flash SSD requires small page sizes. Smaller page sizes are more efficient with less wasted capacity per page. But small page sizes have higher latencies because there are a lot more to keep track of. Large page sizes have lower latencies but sacrifice efficiency and wear life resulting from a greater level of write amplification.

The ZS3 caches all of the writes in DRAM first. Then aggregates them before writing to the SSD cache. This enables ZS3’s Flash SSDs to have large page sizes for faster performance and high efficiency. The DRAM sequentializes the data to each page, thus getting very high page efficiencies; reduced write amplification; longer wear life; and best possible performance automatically.

• **In-Memory Deduplication**

Once again, that large DRAM amount enables Oracle to engineer new areas of automated database performance enhancements. Oracle has placed deduplication in-memory reducing cache consumption by more than 4x while increasing performance by 2x.

• **Parallel Access Sequencing**

Oracle has leveraged its symmetrical multiprocessing (SMP) OS and multi-threaded file and block storage architecture to provide concurrent parallel access to cache. That improved access speeds up cache IO responsiveness by as much as 10x.

• **Infiniband Access**

Oracle has a quad-data-rate (QDR) 40Gbps InfiniBand option on both database servers and ZS3 storage. InfiniBand has built-in remote direct memory access (a.k.a. RDMA). RDMA provides memory-to-memory data transfers at the lowest possible latencies, which is why it is the predominant interconnect for high performance compute clusters (HPC). InfiniBand RDMA reduces the latencies between the database and the storage by more than an order of magnitude. Low latencies always translate into much faster queries and response time.

*How well then does ZS3 auto-tune the database storage for maximum results without breaking the bank or doing a brute force approach? On a scale of 1 to 5 where 5 is the best possible, it appears to be*
approximately a 4.5. Pending the outcome of published benchmark tests, it could go up to a 5 or down to a 4 depending on the actual results. Based on ZFS Storage Appliance 7x20 Series (the previous generation of Application Engineered Storage), and preliminary ZS3 benchmark results, it is most likely to move up to the 5.

- **ZS3 slowing/reducing database storage capacity consumption**

  The ZS3 radically reduces database storage consumption while actually increasing performance. It does it via Hybrid Columnar Compression (HCC), a feature of Oracle Database 11gR2 and later, which is unique to Oracle storage.

  - **HCC**

    HCC provides as much as 50x data compression. HCC is a cooperative partnership between the Oracle Database and Oracle storage. Database columnar data tends to be more similar than other database data. Compression results indicate that Oracle’s Database gets minimally 3 to 5x more data reduction than any other vendor’s best data reduction option, sharply decreasing storage footprint and associated acquisition and operational costs. Even more importantly, because HCC is a cooperative, collaborative database-ZS3 process the compressed data never requires rehydration for reads. This allows the HCC compressed data to be accessed directly so there is no latency/response time/performance degradation ramifications experienced as with other data reduction technologies. Not only is there no degradation, but because the amount of data being read and analyzed is compressed the database queries actually are demonstrably faster ranging from 3x to 8x faster in customer applications.

  - **HCC with Advanced Data Optimization**

    As previously discussed, reducing manual intervention from the database to storage interaction is highly desirable and immensely valued. It alleviates DBA time stressors and ensures a repeatable, optimal outcome. With Oracle Database 12c, DBAs gain access to the Advanced Data Optimization feature which enables them to set policies, based on actual heat map analysis of data activity or inactivity in order to automatically take advantage of HCC at the moment the data is ready. When data has moved from active (read/write with frequent access) to less-active (read mostly) to archive (infrequent access) it can automatically take advantage of the various levels of HCC compression that only Oracle storage achieves.

  How well then does ZS3 slow/reduce database storage capacity consumption? Based on actual Oracle customer results it’s quite impressive. ZS3 receives a definitive 5.

- **ZS3 troubleshooting database storage problems and performance bottlenecks**

  Each ZS3 comes with highly granular, visual analytics.

  - **ZS3 DTrace Analytics**

    Those analytics provide graphical visualization and reporting of the CPU, cache, protocol, drives, memory, networking, and all system related data concurrently. These ZS3 analytics find all devices physical and virtual utilizing the storage and the demands they’re making on the ZS3 storage. It additionally is Oracle database aware, tracking the database’s ZS3 access, hot spots, and bottlenecks. ZS3 DTrace Analytics has demonstrated a much faster ability to troubleshoot problems with quicker deep root cause analysis and understanding. Oracle customers have reported up to 44% faster troubleshooting.

  How well then does ZS3 speed troubleshooting database storage problems and performance bottlenecks? Based on Oracle customer results with previous and current analytics, it does a very nice job. The graphics are intuitive. The only user complaint was that it required a bit of
knowledge/expertise to run effectively. ZS3 receives a 4 here.

- **ZS3 accelerating database backups to stay within backup windows**

  Because each ZS3 is tightly engineered with the Oracle Database, so too is the database’s data protection. This tight engineering enables the ZS3 storage to provide incredibly fast backups to Oracle Databases.

  - **ZS3 + Oracle Database data protection engineering**

    When the Oracle database is storing its data on the ZS3, the ZS3 has the distinct advantage of being fully engineered with the entire database APIs and processes including RMAN, Data Guard, and Flashback. ZS3 snapshots are database aware and make sure that the Oracle Database is properly quiesced (cache flushed and all writes completed in their proper order), before the snapshot is taken. No backup software is required to control the quiescence or the ZS3 snapshot.

    When the ZS3 is being used as a backup target for Oracle Databases that engineered together advantage is demonstrated with the RMAN in sync with the ZFS Backup Appliance. This allows the ZFS Backup Appliance back up the HCC data natively via RMAN without rehydration. That Infiniband RDMA “interconnect” provides a very low latency high bandwidth between the database and the target ZFS Backup Appliance.

    The net results are impressive showing backup performance as high as 25TB/hr. That number is actual real data throughput, not effective throughput. If it were calculated as “effective” throughput or, in other words, the effective amount of data transferred were it not compressed would be the HCC compression ratio times the real throughput performance. This means equivalent throughput could be as high as 1.25PB/hr. Nice to know, but it is better to stick with real actual data throughput. Restore performance is also impressive at an actual throughput rate of 10TB/hr.

  - **How well then does ZS3 accelerate database backups to stay within the backup windows? Based again on production proven Oracle customer (such as Digi-Key), there is nothing faster. ZS3 receives a 5 here.**

- **ZS3 database storage active archiving**

  The ZS3 is cleverly engineered with Oracle database to provide both active and passive archiving. It leverages the best of Oracle Database 12c Advanced Data Optimization (ADO), data usage heat maps, database partitioning, and Flashback Data Archiving to optimize storage efficiency and performance.

  - **Database partitioning and heat maps**

    ZS3 works together with Oracle Database 12c to provide in-database active archiving so the database can use heat maps to direct actively changing data, actively used but unchanging data, and infrequently used data multiple storage pools on the same ZS3 storage system that have different performance, capacity, compression, and availability metrics associated with them.

    Traditionally, large tables can be partitioned easily on a partition key, frequently range-partitioned on the key that represents a time component, with current “active” data located on the higher performance storage pool. As that data ages, becoming less active or “passive”, administrators move it do different partitions. ADO, on the other hand, automatically moves it to partitions on lower cost, lower performing storage pools – eliminating complex, time-consuming, and potentially error prone manual activities. This built-in data “online archiving” is always available to the database and database applications, and can be turned on or off at any time by the DBA – although I’m not quite sure why he or she would turn it off. Data managed through ADO doesn’t have to be recovered or migrated to or from another storage pool or system since it’s still part of the same database. As a result, the hyper-active portion of the database which is receiving all of the updates resides on a smaller and more highly-optimized portion of the ZS3’s storage, benefiting from automatic pruning performed by the database and resulting in much reduced response times. Query access to the archived data is also optimized to
maximize IO throughput which also minimizes access for BIDW type applications. Furthermore, because active data is automatically separated from the archive data that represents the bulk of the database size, active data performance will not degrade as the overall size of the database increases. The engineering together of the ZS3 storage and Oracle Database removes any requirement to regularly archive or purge data in order to maintain the required performance of database applications. The database archive is always online, available at any time through the application, and is also maintained throughout database and application upgrades.

ZS3 storage enables a lot more database data to be kept online at all times for much longer periods of time, greatly improving performance of the applications that depend on and access those large Oracle Database instances.

- **Flashback Data Archiving (FDA)**

All editions (SE1, SE, and EE) of the Oracle Database 12c have at least the basic Flashback Data Archiving feature. Flashback Data Archive (FDA) allows the DBA to store database information for a long time for archiving or restoring purposes, based on a dedicated archive tablespace. Data is still accessed with flashback queries and now the undo data is immediately archived as well in the FDA during the FDA retention period. FDA is used for keeping historical data changes for a predefined retention period such as required by regulatory compliance. FDA also makes it possible to go back in time to get data for comparison purposes. And FDA can recover data after a bad manipulation such as some records accidentally being removed or incorrectly updated.

Oracle’s tight engineering between the ZS3 and Oracle Database 12c enables the FDA to be stored in an extremely compressed state utilizing as few storage resources as is possible while increasing the FDA’s performance.

**How well then does ZS3 accelerate database backups to stay within the backup windows? Based again on production proven Oracle customer there else in the market like it. ZS3 receives a 5 here.**

- **ZS3 management of labor-intensive tasks**

This is another area that the ZS3 really shines due to that tight co-engineering with Oracle Database 12c. The key is the 12c Oracle Intelligent Storage Protocol (OISP), unique to Oracle storage.

- **OISP**

OISP is a unique language that delivers dynamic communications between Oracle Database 12c and Oracle’s ZS3. It utilizes Oracle Database 12c direct NFS (dNFS) and Oracle ZS3’s new OS8®. The key to OISP is that it empowers ZS3 storage to assign resources to specifically optimize Oracle Database 12c performance and efficiency. Remember that logbias and record sizing example earlier. With any other storage system, it would require at least 8 tuning tasks for the logbias and 8 for the record sizing. OISP cuts that down to 2 each or a reduction of over 65% for the DBA.

OISP is engineered to reduce DBA tasks and make their life simpler and easier.

**How well then does ZS3 reduce DBA’s manually labor-intensive task? There is nothing else like it on the market today. Once again the ZS3 receives a top score of 5 here.**

- **ZS3’s ability to reduce TCO for database storage**

Massively less capacity requirements for Oracle Database instances means a lot lower database storage TCO. Exceptional database storage performance management without significant amounts of expensive storage hardware means much lower database storage TCO. Backups and archives that stay compressed within Oracle’s HCC means measurably lower database storage TCO. High degree of automation means much-reduced DBA time and lower database storage TCO. Intuitive visual management reduces DBA problem resolution time and again much lower database storage TCO.

**How well then does ZS3 reduce database storage TCO? There is no question that it’s a lot. How much will vary depending on data, configurations, and individual circumstances. But in the end there is no**
comparable high performance database storage that can match the ZS3 in lowering the TCO and thus receiving the top score of 5.

Summary and Conclusion

Database storage problems are difficult at best and a nightmare at worst. They continue to escalate, as data grows unabated. DBA workarounds attempt to solve these database storage problems but in the end only defer them and in several cases exacerbate other problems (specifically database performance). These workarounds are analogous to treating the symptoms of a disease, but not the root cause.

Oracle’s ZS3 next generation application engineered storage is synergistically engineered with Oracle’s databases to solve these problems at their root cause. And that’s exactly what it does. Oracle’s ZS3 raises the database storage price/performance bar to a level previously unimagined and reduces manual intervention in a manner enthusiastically welcomed by DBAs everywhere.

About the author: Marc Staimer is the founder, senior analyst, and CDS of Dragon Slayer Consulting in Beaverton, OR. The consulting practice of 15 years has focused in the areas of strategic planning, product development, and market development. With over 33 years of marketing, sales and business experience in infrastructure, storage, server, software, databases, and virtualization, he’s considered one of the industry’s leading experts. Marc can be reached at marcstaimer@mac.com.