

Article development led by **acmqueue**
queue.acm.org

New drive technologies and increased capacities create new categories of failure modes that will influence system designs.

BY JON ELERATH

Hard-Disk Drives: The Good, the Bad, and the Ugly

HARD-DISK DRIVES (HDDS) are like the bread in a peanut butter and jelly sandwich—seemingly unexciting pieces of hardware necessary to hold the software. They are simply a means to an end. HDD reliability, however, has always been a significant weak link, perhaps *the* weak link, in data storage. In the late 1980s people recognized that HDD reliability was inadequate for large data storage systems so redundancy was added at the system level with some brilliant software algorithms, and RAID (redundant array of independent disks) became a reality. RAID moved the reliability requirements from the HDD itself to the system of data disks. Commercial implementations of RAID include $n+1$ configurations

such as mirroring, RAID-4 and RAID-5, and the $n+2$ configuration, RAID-6, which increases storage system reliability using two redundant disks (dual parity). Additionally, reliability at the RAID group level has been favorably enhanced because HDD reliability has been improving as well.

Several manufactures produce one-terabyte HDDs and higher capacities are being designed. With higher areal densities (also known as bit densities), lower fly-heights (the distance between the head and the disk media), and perpendicular magnetic recording technology, can HDD reliability continue to improve? The new technology required to achieve these capacities is not without concern. Are the failure mechanisms or the probability of failure any different from predecessors? Not only are there new issues to address stemming from the new technologies, but also failure mechanisms and modes vary by manufacturer, capacity, interface, and production lot.

How will these new failure modes affect system designs? Understanding failure causes and modes for HDDs using technology of the current era and the near future will highlight the need for design alternatives and trade-offs that are critical to future storage systems. Software developers and RAID architects can not only better understand the effects of their decisions, but also know which HDD failures are outside their control and which they can manage, albeit with possible adverse performance or availability consequences. Based on technology and design, where must the developers and architects place the efforts for resiliency?

This article identifies significant HDD failure modes and mechanisms, their effects and causes, and relates them to system operation. Many failure mechanisms for new HDDs remain unchanged from the past, but the insidious undiscovered data corruptions (latent defects) that have plagued all HDD designs to one degree or another will continue to worsen in the near future as areal densities increase.

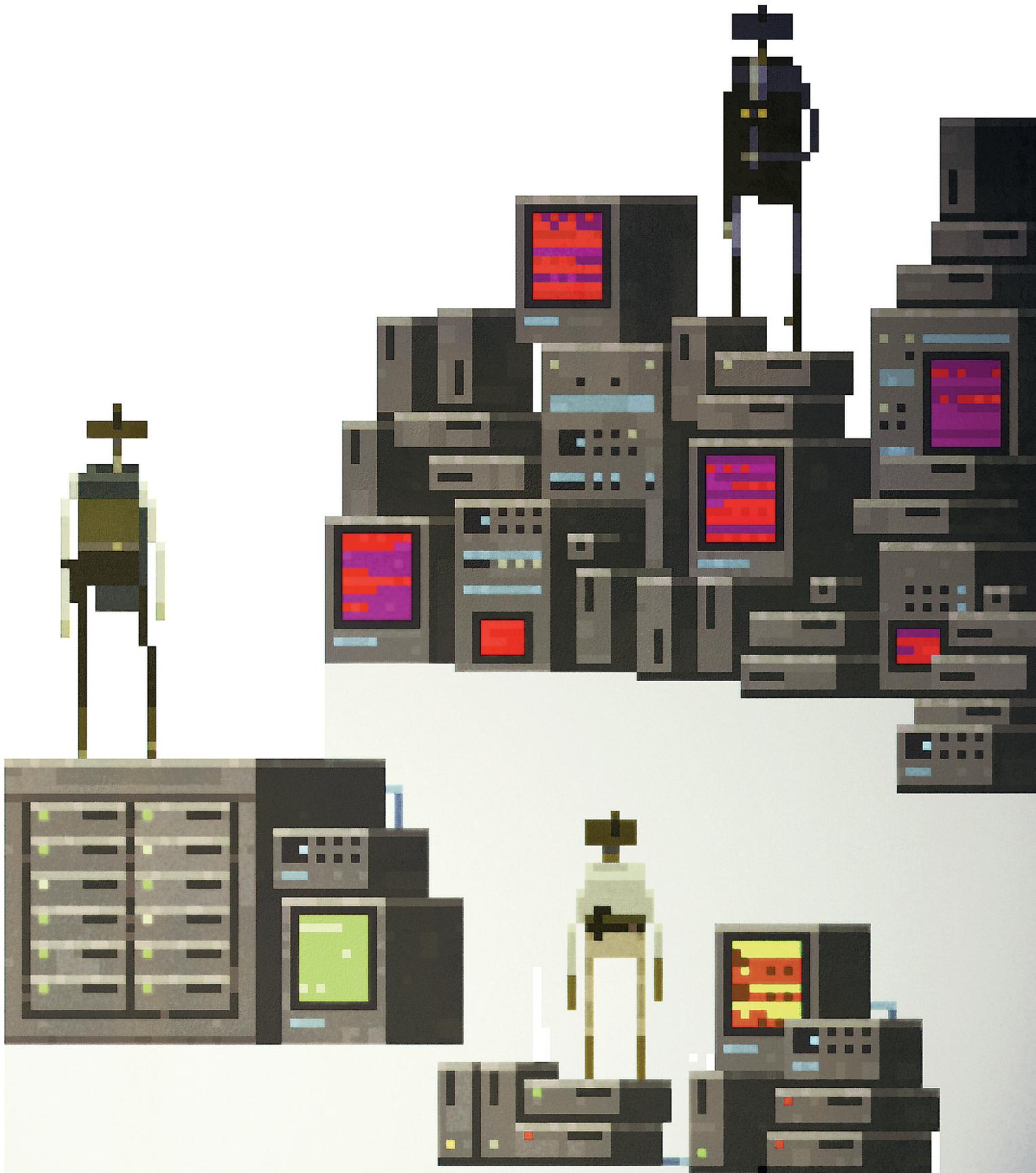
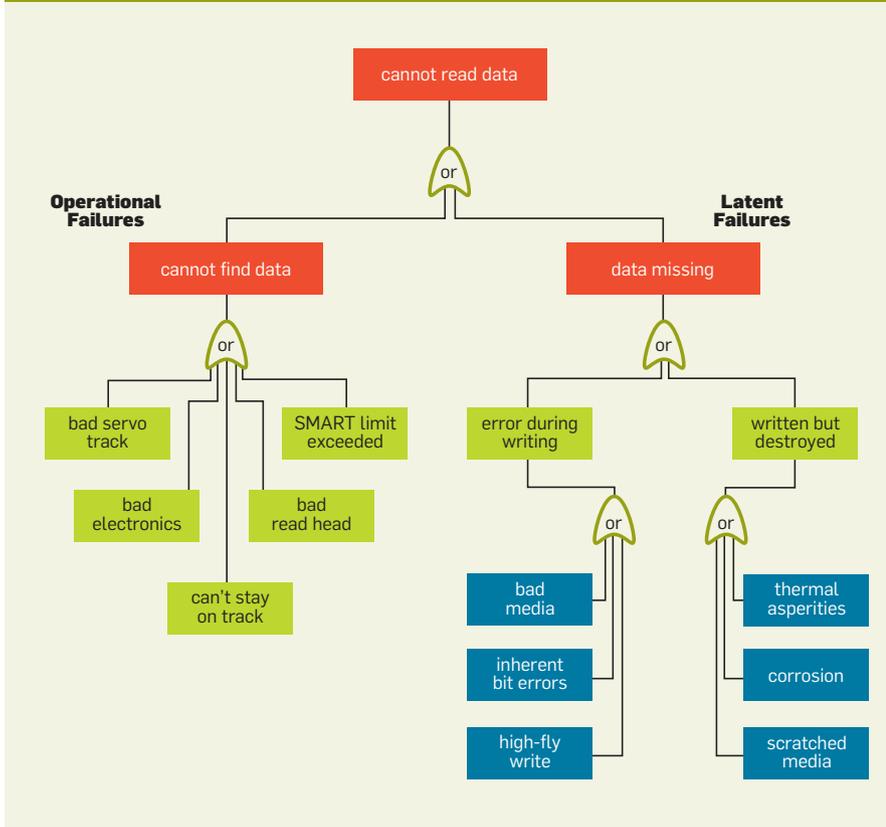


Figure 1: Fault tree for HDD read failures.



Two major categories of HDD failure can prevent access to data: those that fail the entire HDD and those that leave the HDD functioning but corrupt the data. Each of these modes has significantly different causes, probabilities, and effects. The first type of failure, which I term *operational*, is rather easy to detect, but has lower rates of occurrence than the data corruptions or *latent* defects that are not discovered until data is read. Figure 1 is a fault tree for the inability to read data—the topmost event in the tree—showing the two basic reasons that data cannot be read.

Operational Failures: Cannot Find Data

Operational failures occur in two ways: first, data cannot be written to the HDD; second, after data is written correctly and is still present on the HDD uncorrupted, electronic or mechanical malfunction prevents it from being retrieved.

Bad servo track. Servo data is written at regular intervals on every data track of every disk surface. The servo data is used to control the positioning of the read/write heads. Servo data is

required for the heads to find and stay on a track, whether executing a read, write, or seek command. Servo-track information is written only during the manufacturing process and can be neither reconstructed using RAID nor rewritten in the field. Media defects in the servo-wedges cause the HDD to lose track of the heads' locations or where to move the head for the next read or write. Faulty servo tracks result in the inability to access data, even though the data is written and uncorrupted. Particles, contaminants, scratches, or thermal asperities can damage servo data.

Can't stay on track. Tracks on an HDD are not perfectly circular; some are actually spiral. The head position is continuously measured and compared with where it should be. A PES (position error signal) repositions the head over the track. This repeatable run-out is all part of normal HDD head positioning control. NRRO (nonrepeatable run-out) cannot be corrected by the HDD firmware since it is nonrepeatable. Caused by mechanical tolerances from the motor bearings, actuator arm bearings, noise, vibration, and servo-loop response errors, NRRO can

make the head positioning take too long to lock onto a track and ultimately produce an error. This mode can be induced by excessive wear and is exacerbated by high rotational speeds. It affects both ball and fluid-dynamic bearings. The insidious aspect of this type of problem is that it can be intermittent. Specific HDD usage conditions may cause a failure while reading data in a system, but under test conditions the problem might not recur.

Two very interesting examples of inability to stay on track are caused by audible noise. A video file available on YouTube shows a member of Sun's Fishworks team yelling at his disk drives and monitoring the latency in disk operations.⁵ The vibrations from his yelling induce sufficient NRRO that the actuator cannot settle for over 520 ms. While most (some) of us don't yell at our HDDs, vibrations induced by thermal alarms (warning buzzers) have also been noted to induce NRRO and cause excessive latency and time-outs.

SMART limits exceeded. Today's HDDs collect and analyze functional and performance data to predict impending failure using SMART (self-monitoring analysis reporting technology). In general, sector reallocations are expected, and many spare sectors are available on each HDD. If an excessive number occurs in a specific time interval, however, the HDD is deemed unreliable and is failed out.

SMART isn't really that smart. One trade-off that HDD manufacturers face during design is the amount of RAM available for storing SMART data and the frequency and method for calculating SMART parameters. When the RAM containing SMART data becomes full, is it purged, then refilled with new data? Or are the most recent percentages ($x\%$) of data preserved and the oldest $(1-x)\%$ purged? The former method means that a rate calculation such as read-error-rate can be erroneous if the memory fills up during an event that produces many errors. The errors before filling RAM may not be sufficient to trigger a SMART event, nor may the errors after the purge, but had the purge not occurred, the error conditions may easily have resulted in a SMART trip.

In general, the SMART thresholds are set very low, missing numerous

conditions that could proactively fail a HDD. Making the trip levels more sensitive (trip at lower levels) runs the risk of failing HDDs with a few errors that really aren't progressing to the point of failure. The HDD may simply have had a series of reallocations, say, that went smoothly, mapping out the problematic area of the HDD. Integrators must assess the HDD manufacturer's implementation of SMART and see if there are other more instructive calculations. Integrators must at least understand the SMART data collection and analysis process at a very low level, then assess their specific usage pattern to decide whether the implementation of SMART is adequate or whether the SMART decisions need to be moved up to the system (RAID group) level.

Head games and electronics. Most head failures result from changes in the magnetic properties, not electrical characteristics. ESD (electrostatic discharge), high temperatures, and physical impact from particles affect magnetic properties. As with any highly integrated circuit, ESD can leave the read heads in a degraded mode. Subsequent moderate to low levels of heat may be sufficient to fail the read heads magnetically. A recent publication from Google didn't find a significant correlation between temperature and reliability.⁶ In my conversations with numerous engineers from all the major HDD manufacturers, none has said the temperature does not affect head reliability, but none has published a transfer function relating head life to time and temperature. The read element is physically hidden and difficult to damage, but heat can be conducted from the shields to the read element, affecting magnetic properties of the reader element, especially if it is already weakened by ESD.

The electronics on an HDD are complex. Failed DRAM and cracked chip capacitors have been known to cause HDD failure. As the HDD capacities increase, the buffer sizes increase and more RAM is required to cache writes. Is RAID at the RAM level required to assure reliability of the ever-increasing solid-state memory?

Operational Failure Data

In a number of studies on disk failure rates, all mean times between fail-

ures disagree with the manufacturers' specification.^{1-3, 6, 7, 10, 11} More disconcerting are the realizations that the failure rates are rarely constant; there are significant differences across suppliers, and great differences within a specific HDD family from a single supplier. These inconsistencies are further complicated by unexpected and uncontrolled lot-to-lot differences.

In a population of HDDs that are all the same model from a single manufacturer, there can be statistically significant subpopulations, each having a different time-to-failure distribution with different parameters. Analyses of HDD data indicate these subpopulations are so different that they should not be grouped together for analyses because the failure causes and modes are different. HDDs are a technology that defies the idea of "average" failure rate or MTBF; inconsistency is synonymous with variability and unpredictability.

The following are examples of unpredictability that existed to such an extent that at some point in the product's life, these subpopulations dominated the failure rate:

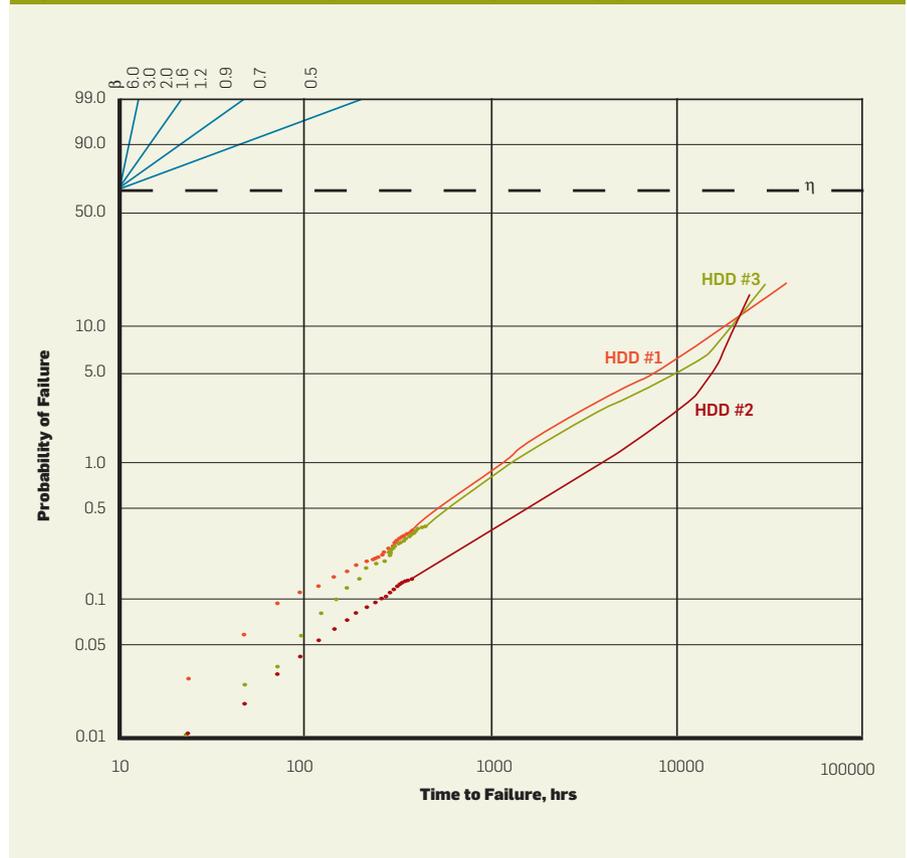
► Airborne contamination. Particles within the enclosure tend to fail HDDs early (scratches and head damage). This can give the appearance of an increasing failure rate. After all the contaminated HDDs fail, the failure rate often decreases.

► Design changes. Manufacturers periodically find it necessary to reduce cost, resolve a design issue discovered late in the test phase, or improve yields. Often, the change creates an improvement in field reliability, but can create more problems than it solves. For example, one design change had an immediately positive effect on reliability, but after two years another failure mode began to dominate and the HDD reliability became significantly worse.

► Yield changes. HDD manufacturers are constantly tweaking their processes to improve yield. Unfortunately, HDDs are so complex that these yield enhancements can inadvertently reduce reliability. Continuous tweaks can result in one month's production being highly reliable and another month being measurably worse.

The net impact of variability in reli-

Figure 2: Weibull time to failure plot for three very different populations.



ability is that RAID designers and software developers must develop logic and operating rules that will accommodate significant variability and the worst-case issues for all HDDs. Figure 2 shows a plot for three different HDD populations. If a straight line were to fit the data points and the slope were 1.0, then the population could be represented by a Weibull probability distribution and have a constant failure rate. (The Weibull distribution is used to create the common bathtub curve.) A single straight line cannot fit either population HDD#2 or HDD#3, so they do not even fit a Weibull distribution. In fact, these do not fit any single closed-form distribution, but are composed of multiple failure distributions from causes that dominate at different points in time. Figure 3 is an example of five HDD vintages from a single supplier. A straight line indicates a constant failure rate; the lower the slope, the more reliable the HDD. A vintage represents a product from a single month.

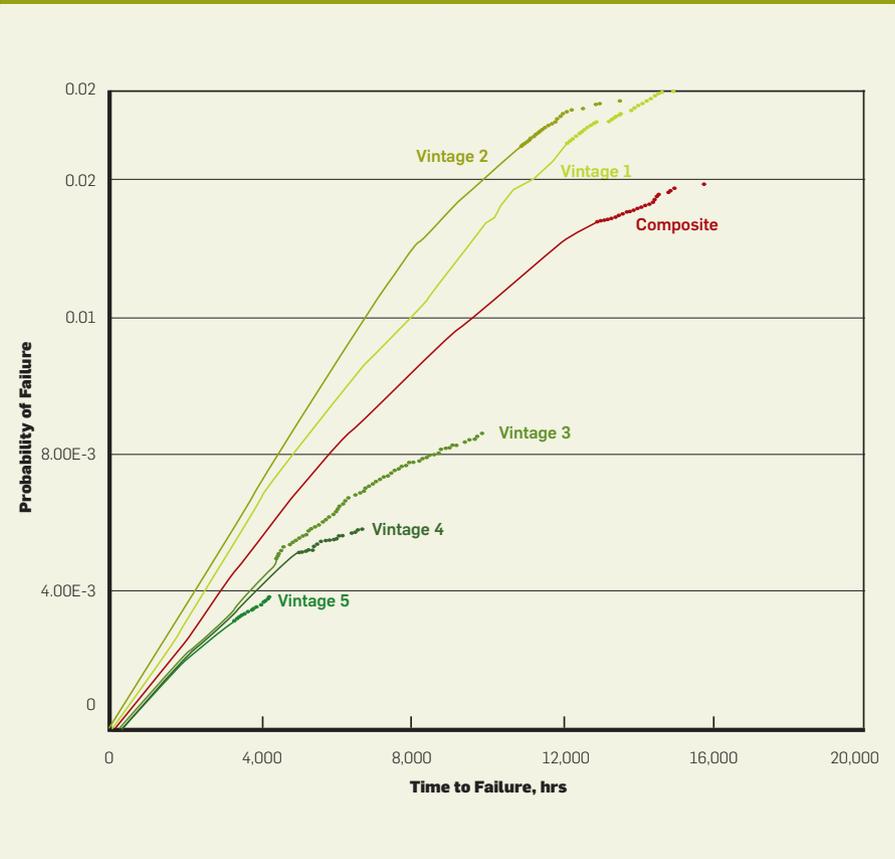
Latent Defects: Data is Corrupted or Missing

The preceding discussion centered on failure modes in which data was good (uncorrupted) but some other electrical, mechanical, or magnetic function was impaired. These modes are usually rather easily detected and allow the system operator to replace the faulty HDD, reconstruct data on the new HDD, and resume storage functions. But what about data that is missing or corrupted because it either was not written well initially or was erased or corrupted *after* being written well. All errors resulting from missing data are latent because the corrupted data is resident without the knowledge of the user (software). The importance of latent defects cannot be overemphasized. The combination of a latent defect followed by an operational failure is the most likely sequence to result in a double failure and loss of data.¹

To understand latent defects better, consider the common causes.

Write errors can be corrected using a read-verify command, but these require an extra read command after writing, and can nearly double the effective time to write data. The BER (bit-error rate) is a statistical measure

Figure 3: Failure rate over time for five vintages and the composite.



of the effectiveness of all the electrical, mechanical, magnetic, and firmware control systems working together to write (or read) data. Most bit errors occur on a read command and are corrected using the HDD's built-in error-correcting code algorithms, but errors can also occur during writes. While BER does account for some fraction of defective data, a greater source of data corruption is the magnetic recording media coating the disks.

The distance that the read-write head flies above the media is carefully controlled by the aerodynamic design of the slider, which contains the reader and writer elements. In today's designs, the fly height is less than 0.3 μ -in. Events that disturb the fly height, increasing it above the specified height during a write, can result in poorly written data because the magnetic-field strength is too weak. Remember that magnetic-field strength does not decrease linearly as a function of distance from the media, but is a power function, so field strength falls off very rapidly as the distance between the head and media increases. Writing data while

the head is too high can result in the media being insufficiently magnetized so it cannot be read even when the read element is flying at the specified height. If writing over a previously written track, the old data may persist where the head was flying too high. For example, if all the HDDs in a cabinet are furiously writing at the same time, self-induced vibrations and resonances can be great enough to affect the fly height. Physically bumping or banging an HDD during a write or walking heavily across a poorly supported raised floor can create excessive vibration that affects the write.

A more difficult problem to solve is persistent increase in the fly height caused by buildup of lubrication or other hydrocarbons on the surface of the slider. Hydrocarbon lubricants are used in three places within enclosed HDDs. To reduce the NRRO, motors often use fluid-dynamic bearings. The actuator arm that moves the heads pivots using an enclosed bearing cartridge that contains a lubricant. The media itself also has a very thin layer of lubricant applied to prevent the

heads from touching the media itself. Lubricants on the media can build up on the head under certain circumstances and cause the head to fly too high. Lube buildup can also mean that uncorrupted, well-written data cannot be read because the read element is too far from the media. Lube buildup can be caused by the mechanical properties of the lubricant, which is dependent upon the chemical composition. Persistent high fly height can also be caused by specific operations. For example, when not writing or reading, if the head is left to sit above the same track while the disks spin, lubricant can collect on the heads. In some cases simply powering down the HDD will cause the heads to touch down (as they are designed to do) in the landing zone to disturb the lube buildup. This is very design specific, however, and does not always work.

During the manufacturing process, the surface of the HDD is checked and defects are mapped out, and the HDD firmware knows not to write in these locations. They also add “padding” around the defective area mapping out more blocks than the estimated minimum, creating additional physical distance around the defect that is not available for storing data. Since it is difficult to determine the exact length, width, and shape of a defect, the added padding provides an extra safeguard against writing on a media defect.

Media imperfections such as voids (pits), scratches, hydrocarbon contamination (various oils), and smeared soft particles can not only cause errors during writing, but also corrupt data after it has been written. The sputtering process used to apply some of the media layers can leave contaminants buried within the media. Subsequent contact by the slider can remove these bumps, leaving voids in which the media is defective. If data is already written there, the data is corrupted. If none is written, the next write process will be unsuccessful, but the user won’t know this unless a write-verify command is used.

Early reliability analyses assumed that once written, data will remain undestroyed except by degradation of the magnetic properties of the media, a process known as bit-rot. Bit-rot, in



Based on technology and design, where must the developers and architects place the efforts for resiliency?



which the magnetic media is not capable of holding the proper magnetic field to be correctly interpreted as a 0 or a 1, is really not an issue. Media can degrade, but the probability of this mode is inconsequential compared with other modes. Data can become corrupted whenever the disks are spinning, even when data is not being written to or read from the disk. Common causes for erasure include thermal asperities, corrosion, and scratches or smears.

Thermal asperities are instances of high heat for a short duration caused by head-disk contact. This is usually the result of heads hitting small “bumps” created by particles that remain embedded in the media surface even after burnishing and polishing. The heat generated on a single contact can be high enough to erase data. Even if not on the first contact, cumulative effects of numerous contacts may be sufficient to thermally erase data or mechanically destroy the media coatings and erase data.

The sliders are designed to push away airborne particles so they do not become trapped between the head and disk surface. Unfortunately, removing all particles that are in the 0.3 μ -in. range is very difficult, so particles do get caught. Hard particles used in the manufacture of an HDD, such as Al_2O_3 , TiW, and C, will cause surface scratches and data erasure. These scratches are then media defects that are not mapped out, so the next time data is written to those locations the data will be corrupted immediately. Other “soft” materials such as stainless steel can come from assembly tooling and aluminum from residuals from machining the case. Soft particles tend to smear across the surface of the media rendering the data unreadable and unwritable. Corrosion, although carefully controlled, can also cause data erasure and may be accelerated by high ambient heat within the HDD enclosure and the very high heat flux from thermal asperities.

Latent Defects Data

Latent defects are the most insidious kinds of errors. These data corruptions are present on the HDD but undiscovered until the data is read. If no operational failures occur at the first

reading of the data, the corruption is corrected using the parity disk and no data is lost. If one HDD, however, has experienced an operational failure and the RAID group is in the process of reconstruction when the latent defect is discovered, that data is lost. Since latent defects persist until discovered (read) and corrected, their rate of occurrence is an extremely important aspect of RAID reliability.

One study concludes that the BER is fairly inconsequential in terms of creating corrupted data,⁴ while another claims the rate of data corruption is five times the rate of HDD operating failures.⁸ Analyses of corrupted data identified by specific SCSI error codes and subsequent detailed failure analyses show that the rate of data corruption for all causes is significant and must be included in the reliability model.

NetApp (Network Appliance) completed a study in late 2004 on 282,000 HDDs used in RAID architecture. The RER (read-error rate) over three months was 8×10^{-14} errors per byte read. At the same time, another analysis of 66,800 HDDs showed an RER of approximately 3.2×10^{-13} errors per byte. A more recent analysis of 63,000 HDDs over five months showed a much-improved 8×10^{-15} errors per byte read. In these studies, data corruption is verified by the HDD manufacturer as an HDD problem and not a result of the operating system controlling the RAID group.

While Jim Gray of Microsoft Research asserted that it is reasonable to transfer 4.32×10^{12} bytes/day/HDD, the study of 63,000 HDDs read 7.3×10^{17} bytes of data in five months, an approximate read rate of 2.7×10^{11} bytes/day/HDD.⁴ Using combinations of the

RERs and number of bytes read yields the hourly read failure rates shown in the table here.

Latent defects do not occur at a constant rate, but in bursts or adjacent physical (not logical) locations. Although some latent defects are created by wear-out mechanisms, data is not available to discern wear-out from those that occur randomly at a constant rate. These rates are between 2 and 100 times greater than the rates for operational failures.

Potential Value of Data Scrubbing

Latent defects (data corruptions) can occur during almost any HDD activity: reading, writing, or simply spinning. If not corrected, these latent defects will result in lost data when an operational failure occurs. They can be eliminated, however, by background scrubbing, which is essentially preventive maintenance on data errors. During scrubbing, which occurs during times of idleness or low I/O activity, data is read and compared with the parity. If they are consistent, no action is taken. If they are inconsistent, the corrupted data is recovered and rewritten to the HDD. If the media is defective, the recovered data is written to new physical sectors on the HDD and the bad blocks are mapped out.

If scrubbing does not occur, the period of time to accumulate latent defects starts when the HDD begins operation in the system. Since scrubbing requires reading and writing data, it can act as a time-to-failure accelerator for HDD components with usage-dependent time-to-failure mechanisms. The optimal scrub pattern, rate, and time of scrubbing is HDD-specific and must be determined in conjunction with the HDD manufacturer to assure

that operational failure rates are not increased.

Frequent scrubbing can affect performance, but too infrequent scrubbing makes the $n+1$ RAID group highly susceptible to double disk failures. Scrubbing, as with full HDD data reconstruction, has a minimum time to cover the entire HDD. The time to complete the scrub is a random variable that depends on HDD capacity and I/O activity. The operating system may invoke a maximum time to complete scrubbing.

Future Technology and Trade-Offs

How are those failure modes going to impact future HDDs that have more than one-terabyte capacity? Certainly, all the failure mechanisms that occur in the 1TB drive will persist in higher density drives that use perpendicular magnetic recording (PMR) technology. PMR uses a “thick,” somewhat soft underlayer making it susceptible to media scratching and gouging. The materials that cause media damage include softer metals and compositions that were not as great a problem in older, longitudinal magnetic recording. Future higher density drives are likely to be even more susceptible to scratching because the track width will be narrower.

Another PMR problem that will persist as density increases is side-track erasure. Changing the direction of the magnetic grains also changes the direction of the magnetic fields. PMR has a return field that is close to the adjacent tracks and can potentially erase data in those tracks. In general, the track spacing is wide enough to mitigate this mechanism, but if a particular track is written repeatedly, the probability of side-track erasure increases. Some applications are optimized for performance and keep the head in a static position (few tracks). This increases the chances of not only lube buildup (high fly writes) but also erasures.

One concept being developed to increase bit-density is heat assisted magnetic recording (HAMR).⁹ This technology requires a laser within the write head to heat a very small area on the media to enable writing. High-stability media using iron-platinum alloys allow bits to be recorded on much

Range of average read error rates.

		Bytes Read per Hour	
		Low rate (1.35×10^9)	High rate (1.35×10^{10})
Read Errors per Byte per HDD	Low (8.0×10^{-15})	1.08×10^{-5} err/hr	1.08×10^{-4} err/hr
	Medium (8.0×10^{-14})	1.08×10^{-4} err/hr	1.08×10^{-3} err/hr
	High (3.2×10^{-13})	4.32×10^{-4} err/hr	4.32×10^{-3} err/hr

smaller areas than today's standard media without being limited by super-paramagnetism. Controlling the amount and location of the heat are, of course, significant concerns.

RAID is designed to accommodate corrupted data from scratches, smears, pits, and voids. The data is re-created from the parity disk and the corrupted data is reconstructed and rewritten. Depending on the size of the media defect, this may be a few blocks or hundreds of blocks. As the areal density of the HDDs increases, the same physical size of the defect will affect more blocks or tracks and require more time for re-creation of data. One trade-off is the amount of time spent recovering corrupted data. A desktop HDD (most ATA drives) is optimized to find the data no matter how long it takes. In a desktop there is no redundancy and it is (correctly) assumed that the user would rather wait 30–60 seconds and eventually retrieve the data than to have the HDD give up and lose data.

Each HDD manufacturer has a proprietary set of recovery algorithms it employs to recover data. If the data cannot be found, the servo controller will move the heads a little to one side of the nominal center of the track, then to the other side. This off-track reading may be performed several times at different off-track distances. This is a very common process used by all HDD manufacturers, but how long can a RAID group wait for this recovery?

Some RAID integrators may choose to truncate these steps with the knowledge that the HDD will be considered failed even though it is not an operational failure. On the other hand, how long can a RAID group response be delayed while one HDD is trying to recover data that is readily recoverable using RAID? Also consider what happens when a scratch is encountered. The process of recovery for a large number of blocks, even if the process is truncated, may result in a time-out condition. The HDD is off recovering data or the RAID group is reconstructing data for so long that the performance comes to a halt; a time-out threshold is exceeded and the HDD is considered failed.

One option is quickly to call the offending HDD failed, copy all the data

to a spare HDD (even the corrupted data), and resume recovery. A copy command is much quicker than reconstructing the data based on parity, and if there are no defects, little data will be corrupted. This means that reconstruction of this small amount of data will be fast and not result in the same time-out condition. The offending HDD can be (logically) taken out of the RAID group and undergo detailed diagnostics to restore the HDD and map out bad sectors.

In fact, a recent analysis shows the true impact of latent defects on the frequency of double disk failures.¹ Early RAID papers stated that the only failures of concern were operational failures because, once written, data does not change except by bit-rot.

Improving Reliability

Hard-disk drives don't just fail catastrophically. They may also silently corrupt data. Unless checked or scrubbed, these data corruptions result in double disk failures if a catastrophic failure also occurs. Data loss resulting from these events is the dominant mode of failure for an $n+1$ RAID group. If the reliability of RAID groups is to increase, or even keep up with technology, the effects of undiscovered data corruptions must be mitigated or eliminated. Although scrubbing is one clear answer, other creative methods to deal with latent defects should be explored.

Multi-terabyte capacity drives using perpendicular recording will be available soon, increasing the probability of both correctable and uncorrectable errors by virtue of the narrowed track widths, lower flying heads, and susceptibility to scratching by softer particle contaminants. One mitigation factor is to turn uncorrectable errors into correctable errors through greater error-correcting capability on the drive (4KB blocks rather than 512- or 520-byte blocks) and by using the complete set of recovery steps. These will decrease performance, so RAID architects must address this trade-off.

Operational failure rates are not constant. It is necessary to analyze field data, determine failure modes and mechanisms, and implement corrective actions for those that are most problematic. The operating system

should consider optimizations around these high-probability events and their effects on the RAID operation.

Only when these high-probability events are included in the optimization of the RAID operation will reliability improve. Failure to address them is a recipe for disaster. ■

Related articles on queue.acm.org

You Don't Know Jack about Disks

Dave Anderson

<http://queue.acm.org/detail.cfm?id=864058>

CTO Roundtable: Storage

<http://queue.acm.org/detail.cfm?id=1466452>

A Conversation with Jim Gray

<http://queue.acm.org/detail.cfm?id=864078>

References

1. Elerath, J.G. Reliability model and assessment of redundant arrays of inexpensive disks (RAID) incorporating latent defects and non-homogeneous poisson process events. Ph.D. dissertation, Department of Mechanical Engineering, University of Maryland, 2007.
2. Elerath, J.G. and Pecht, M. Enhanced reliability modeling of RAID storage systems. In *Proceedings of the 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, (Edinburgh, UK, June 2007).
3. Elerath, J.G. and Shah, S. Server class disk drives: How reliable are they? In *Proceedings of the Annual Reliability and Maintainability Symposium*, (January 2004), 151–156.
4. Gray, J. and van Ingen, C. Empirical measurements of disk failure rates and error rates. Microsoft Research Technical Report, MSR-TR-2005-166, December 2005.
5. Gregg, B. Shouting in the datacenter, 2008; <http://www.youtube.com/watch?v=tDacjrSCeq4>.
6. Pinheiro, E., Weber, W.-D., and Barroso, L.A. Failure trends in a large disk drive population. In *Proceedings of the Fifth Usenix Conference on File and Storage Technologies (FAST)*, (February 2007).
7. Schroeder, B. and Gibson, G. Disk failures in the real world: What does an MTTF of 1,000,000 hours mean to you? In *Proceedings of the Fifth Usenix Conference on File and Storage Technologies (FAST)*, (February 2007).
8. Schwarz, T.J.E., et al. Disk scrubbing in large archival storage systems. In *Proceedings of the IEEE Computer Society Symposium* (2004), 1161–1170.
9. Seigler, M. and McDaniel, T. What challenges remain to achieve heat-assisted magnetic recording? *Solid State Technology* (Sept. 2007); http://www.solid-state.com/display_article/304597/5/ARTCL/none/none/What-challenges-remain-to-achieve-heat-assisted-magnetic-recording?/.
10. Shah, S. and Elerath, J.G. Disk drive vintage and its affect on reliability. In *Proceedings of the Annual Reliability and Maintainability Symposium*, (January 2004), 163–167.
11. Sun, F. and Zhang, S. Does hard-disk drive failure rate enter steady-state after one year? In *Proceedings of The Annual Reliability and Maintainability Symposium*, IEEE, (January 2007).

Jon Elerath is a staff reliability engineer at SolFocus. He has focused on hard-disk drive reliability for more than half his 35-plus-year career, which includes positions at NetApp, General Electric, Tegal, Tandem Computers, Compaq, and IBM.