Improving Availability in VERITAS Environments

Karl Larson – Tellme Networks
Todd Stansell – GNAC

ABSTRACT

Demands for high availability are increasing almost as fast as storage and performance requirements, posing seemingly impossible challenges for system administrators. VERITAS provides a variety of tools aimed at overcoming these obstacles, but they're not often used effectively.

We'll show you some tools and techniques that work with VERITAS products to help you evaluate whether your systems are already overloaded, and what you can do to stall for time. We'll describe some environments particularly prone to storage-scaling issues and suggest changes you should make before crisis strikes. Finally, we'll provide some suggestions for decreasing the consequences of failures not preventable through fault tolerance.

How To Tell If You're Already Screwed, And Some Quick Fixes

Gather Layout Information

In order to make intelligent changes to improve your storage environment, you must collect detailed performance and configuration information. Most of the tools needed to collect this information are probably already installed on your systems – you just need to use them. We'll also mention a few other tools and scripts that we and others have created to collect and present this data. Details on how to obtain these third-party tools are included in the appendix.

The most important initial steps with VERITAS Volume Manager (VxVM) installations are obtaining information about and understanding your disk layout. The provided utilities do a poor job of helping you visualize your overall layout, so it's essential that you generate your own diagram, even if it's only on paper. It's important that you include the layout of the underlying storage if you're using some type of hardware RAID device.

The vxprint utility is usually your best source for storage layout information, although it takes time to understand what it means. You should also consider using a tool like "save-vxlayout" to regularly get a copy of this off of the local system for disaster recovery purposes. If you're using EMC storage arrays, you might also consider using emcprint, a script which adds physical disk information to the vxprint output.

Collect Performance Statistics

It's also vital to understand your storage performance characteristics before making significant changes. Again, you ideally want to include performance statistics from underlying storage. All of you should be familiar with the standard Solaris disk performance utility, iostat. Most of you have probably also seen vxstat, which is included with VERITAS Volume Manager and allows you to get similar performance numbers about logical objects managed by Volume Manager. VERITAS has other performance analysis tools available internally that you may be able to convince them to share, including vxfssar, which provides extensive VERITAS File System (VxFS) performance statistics.

However, when you're using hardware RAID, these commands don't provide any insight into how the underlying components are behaving. For instance, you really need to know when cache hit rates start to fall off, particularly with write cache. With EMC Symmetrix arrays, you need to install the additional package called symcli. With Clariion products, use navicli getcache. You may need to run navicli setstats -on if you see that statistics are not enabled.

Presenting these statistics meaningfully is more challenging, but a couple of tools make this process easier. Cricket is a good way of doing trends-based monitoring of more than just network equipment. We've found it useful for tracking disk performance, filesystem usage, network utilization, etc. By allowing you to create views of all of these statistics together, Cricket makes it much easier to see when you are starting to hit a bottleneck in one of them. However, when all you want to do is collect some quick performance statistics from Volume Manager, you might give vxstat2gnuplot a try.

You should also take a quick look at your backup and restore times. In growing environments, it's critical to know how long restores will take and whether this is acceptable to your management. We've found that in certain environments, incremental restores, for instance, can take many times longer than full restores.

Looking At What You've Assembled

Trends

Detailed graphs of system performance and utilization can be crucial for several purposes, with capacity planning probably being the most obvious one. Assuming linear growth, you can easily determine when you will need to buy disks, add servers, or move users to a new machine. In other words, it shows you roughly how long you have before you reach capacity. You can sometimes also use these graphs to determine what that capacity is, once you have an example of a system that is overloaded.

These graphs can also provide an alarm system of sorts for problems you've never run across before. The most common examples of this are sudden changes in utilization, such as those shown in the graph of interface utilization (Figure 1).

Peak usage times

In order to optimize system performance and availability, you need to understand when your peak usage occurs and how extreme it is. These peaks often occur at somewhat different times than your management may believe, so be sure to collect your own data to see for yourself, possibly using cricket or vxstat2gnuplot, as mentioned above.

In many cases, you may find that you have multiple peaks, possibly corresponding to users in different time zones. With careful planning, it's possible to use your knowledge of these peaks to share resources more effectively. In the example above (Figure 2), you can see how sharing storage more effectively between users with different usage patterns might allow you to more fully utilize your storage.

Of course, sharing resources with users around the world can also reduce off-peak periods, which you may have relied on for maintenance or to improve backup performance.

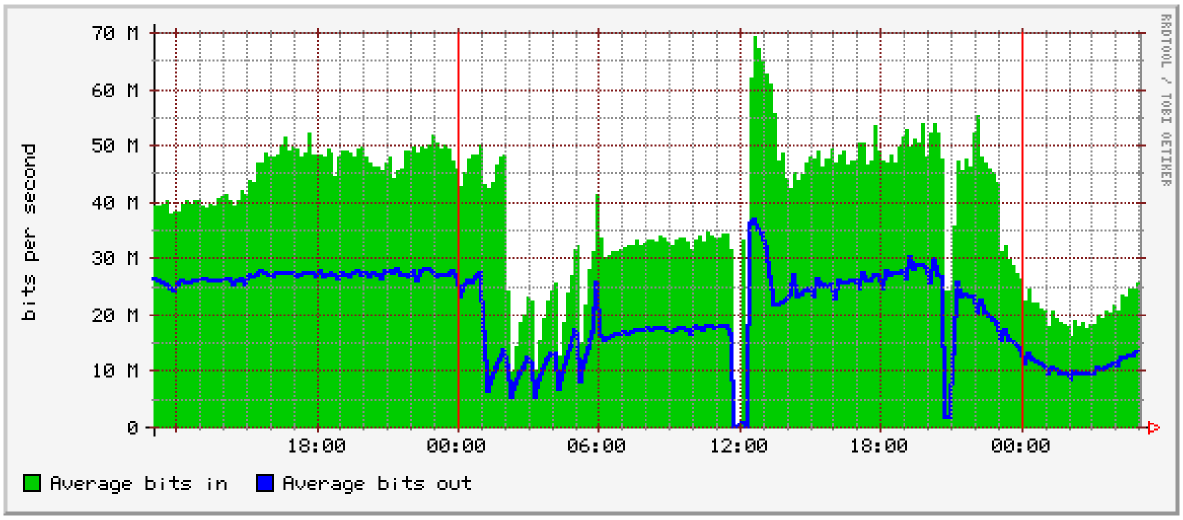

Figure 1: Example of Cricket data used to understand anomalous trends.

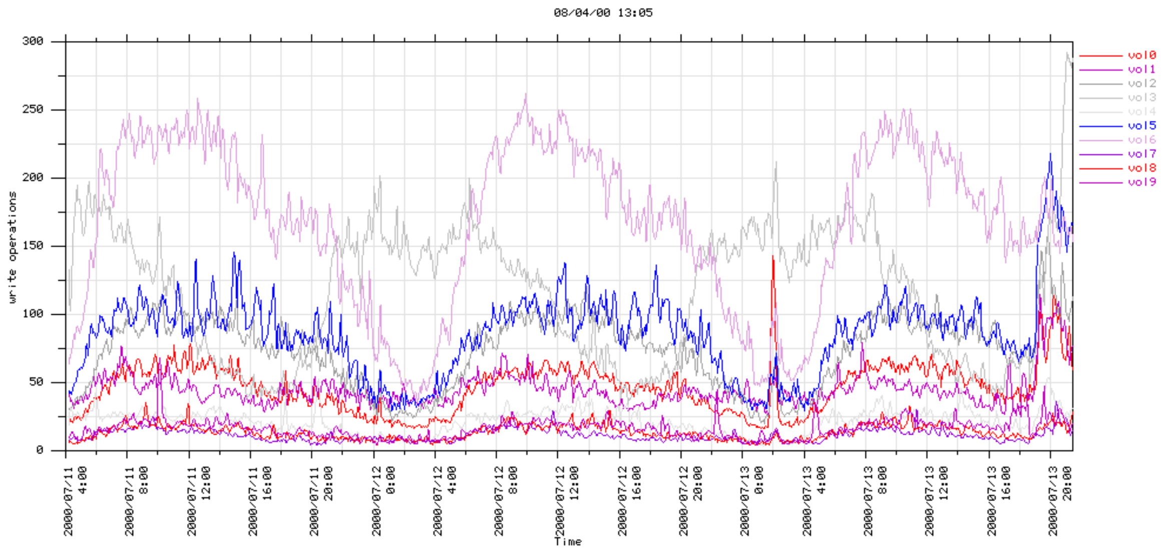

Figure 2: Example of vxstat2gnuplot graph.

Hardware RAID Write Cache Hit Rate ≠ 100%

This typically means that the write cache is overflowing. In other words, the physical disks are not able to keep up with the data being written to them, and even an extremely large write cache will eventually fill. Under these circumstances, write service times (average write times in vxstat) can quickly increase to over ten times their normal value.

Adding additional cache is usually of limited benefit, since the writes still must be committed to disk, and will eventually fill it. The best approach is to spread busy volumes across more disks, which on some systems might prevent you from using the full capacity of each drive. Upgrading to faster drives can also be of some benefit.

Disk Read/Write Times Frequently Exceed 100ms

It's hard to provide exact thresholds that indicate disk performance problems, however access times in excess of 100ms almost always indicate that something is wrong. One thing to look for is a few disks with significantly higher access times than most others on a system. In some cases, access times as low as 20ms may be a sign of trouble.

Reducing The Impact of the More Troublesome Single Points of Failure

Careful storage management and hardware selection can eliminate most of the causes of long sustained outages, but OS, application, non-redundant hardware, and mischievous users remain as sources of occasional system crashes and downtime. The following frequently result from these situations and, in some cases, lead to each other.

Filesystem corruption

When filesystems are not unmounted cleanly, it is required that they be checked for consistency. This process usually occurs very rapidly with VxFS filesystems because all pending changes have been tracked via intent logs, and thus, only those changes need to be validated. Anything that does not appear in the intent logs is assumed to be consistent. With VxFS filesystems, there are four states that the filesystem can be in during a check.

1. The clean bit is set, so no check is required.
2. The log replay is successful, so the filesystem is marked clean.
3. The log replay is unsuccessful, requiring a full check.
4. The log replay is successful, but corruption is later detected.

When a full filesystem check is necessary, as in case three, it can be extremely time consuming. We've seen this take as long as 24 hours on a 500 gigabyte filesystem containing over 50 million files. The only way to reduce the time a full filesystem check requires is to create smaller filesystems. You should determine the maximum acceptable time your filesystems can be down in the event of a full check and size them appropriately. If you have multiple filesystems that require a full check, be sure to do them in parallel to reduce the duration of the outage.

As for the fourth case, it should never occur. However, if it does, you will find out what a metasave is – hopefully you never will.

Volume Synchronization

Volume Manager's dirty region logging (DRL) provides logging features similar to those provided by filesystem intent logs. DRL is intent logs on the volume block level. It protects you against the need to resynchronize mirrored or RAID-5 plexes after a shutdown where the volume could not be marked clean. As with filesystem intent logs, there is still the chance that a full resynchronization is required, but it is possible to decrease the time and performance impact of this.

Volume resynchronization can take many times longer to complete on an active volume. Consider waiting for resynchronization to complete, or nearly complete before restarting services. You can see the progress of Volume Manager tasks, like resynchronizations, with the `vxtask` command in version 3.x. With version 2.x, you can watch resynchronization progress with `vxstat -f b` that will show atomic copies, which are copies between plexes.

When a mirror or a volume snapshot is being attached to an already mirrored volume, you can reduce the performance impact by setting a preferred plex. By telling Volume Manager to prefer one plex over another, it will perform all read operations from the preferred plex only. This is important because when a third plex is being attached to a volume, it will synchronize from only one of the existing plexes. If you force all other reads to occur from the remaining plex, overall performance of the volume should improve. You can determine which existing plex is being used to synchronize the new plex by using `vxstat -f b` and look at which disks are performing the atomic reads.

Damaged Data Due To User or Application Error

Even if you manage to protect against corruption due to system crashes, you still have to contend with damage caused by rogue users and applications. The traditional approach to protecting against this relies on regular backups. Unfortunately, restoring from backup usually means that you lose up to a day's worth of changes, and these restores can often take a long time to complete. A couple of VERITAS technologies allow you to recover from this type of failure much more quickly: Volume Manager snapshots, VxFS snapshots, and VxFS checkpoints.

A Volume Manager snapshot is essentially a write-only mirror of the volume that is broken off and becomes its own volume. The only performance impact on the original volume is during the initial synchronization. Another disadvantage is that it requires as much disk space as a full mirror of the original volume. Since it becomes its own volume, it assumes all of the characteristics of a normal volume including persistence across reboots. However, it is created with the nolog option set, so you will need to stop the volume and change this parameter before you attempt to associate a logging plex. At any time, you can mount this in place of the original filesystem to revert all changes or mount it and copy individual files that need to be restored. A Volume Manager snapshot is a great way to take a backup of your data with minimal impact to the primary filesystem.

A VxFS snapshot is mounted by associating storage with a mounted VxFS filesystem. It reflects the state of the original VxFS filesystem (known as the snapped filesystem) at the time the snapshot was created. As the snapped filesystem changes, the original version of those changed blocks gets copied to the snapshot. The main advantage of a VxFS snapshot is that it uses less disk space than a Volume Manager snapshot, only requiring enough storage to accommodate the blocks that change during the time the snapshot is mounted. Unfortunately, it can be hard to accurately predict how much disk space those changes will require – the snapshot becomes disabled when the underlying storage fills up. Filesystem performance can be significantly impacted while the snapshot is mounted, since three operations are required for each change. Also, since a snapshot can only be associated with a mounted filesystem, it is not persistent across reboots. For these reasons, it is typically only used as a way of providing a consistent image during a backup, but they don't provide a mechanism for reverting the entire filesystem.

VxFS checkpoints are a sophisticated way of providing the Volume Manager snapshot capability of reverting a filesystem back to a specific point in time nearly instantaneously. They work in much the same way a VxFS snapshot does, but they are not mounted and they use the primary storage to store the changed blocks. They are useful for performing backups using the Block Level Incremental Backup product. Since you can't mount a VxFS checkpoint, you can't easily restore an individual file.

Failed Components

Volume manager does an excellent job of providing details when errors occur. It is your job to make sure those notifications go to the right people. Vxnotify is used to provide email notification of volume manager errors. It defaults to sending those notifications to root on the local machine, but you can specify any email address to receive these notifications in the startup script, /etc/init.d/vxvm-recover, by changing the argument it passes to vxrelocd. Volume Manager also reports errors to syslog, providing another mechanism for failure notification. Syslog is the only mechanism by which VxFS reports errors, so it is critical to monitor syslog in some fashion, such as with Netcool.

Full Filesystems

Full VxFS filesystems can be even more troublesome than UFS filesystems. In both cases, it is possible to grow the filesystem online. However, VxFS filesystems need space for additional structural information during filesystem growth operations, making it impossible to grow a completely full filesystem.

To prevent this situation from occurring, always monitor your filesystems carefully. If you aren't completely confident in your monitoring system, you should at least create a ˜10mb placeholder file so that you can easily free up just enough space to grow the filesystem. Growing a filesystem by a large amount requires more space for structural information than growing it by a small amount, so it's sometimes necessary to grow it in multiple steps. However, we've been warned by folks at VERITAS that it's safer to limit the number of growth operations.

Failover

Any application or service that maintains unique, persistent data requires some sort of recovery strategy. Depending on business needs, it may be acceptable to sustain a few unplanned 30-60 minute outages per year, with a chance of a multiple hour outage. If this is not acceptable, some sort of failover scheme is necessary. Automated failover becomes even more critical if you're committed to providing such a high level of service availability that even occasional planned downtime is otherwise impossible.

A failover strategy can be as simple as having an idle system connected to the same disk array. This sort of primitive approach requires thorough rehearsal and process documentation, but it can be nearly as effective as more costly solutions.

A step up from this is what amounts to homegrown automation, or partial automation. A fairly simple script can be used to ping another machine that's running a service, and if it's not there, start that service locally. However, it can be tricky to accurately detect every possible failure mode, and the failover of persistent storage is hard to automate safely.

VERITAS Cluster Server is one of the more popular commercial solutions for automating failovers. This may be the easiest, most reliable solution if you're running one of the applications or services for which they provide a bundled solution, such as Oracle or NFS. It automates things so your failover is consistent. It understands dependencies between services to ensure consistent adherence to failover policies. It parallelizes many activities, such as checking and mounting filesystems, which results in rapid startup times. Finally, it has sophisticated safeguards to protect against what's generally called split-brain syndrome. Split-brain occurs when two hosts try to write to the same disk, often resulting in corruption.

Characteristics of Environments Likely to Experience Extreme Scalability Problems and What You Should Be Fixing Now

Lots of Small Files

Metadata updates can be bottleneck

The VERITAS File System stores all of its metadata at the beginning of the volume, which includes things like directory information and the intent log. Certain types of filesystems, particularly those with a large number of small files, can tend to be performance bound by the speed of the disk(s) on which the first portion of the filesystem is located. The performance of intent log writes is reduced because the filesystem is also performing other operations. In some cases, this can be improved considerably by moving the intent logs onto dedicated storage, which is what the QuickLog product allows you to do. It does add considerable complexity, particularly in clustered environments, but the performance gain is often worth the hassle.

In some cases, vxstat may indicate that even after the use of QuickLog, the disk(s) that contain the first portion of the filesystem are having a hard time keeping up. Intent logs aren't the only type of metadata normally stored at the beginning of a VxFS filesystem, but they are the only type that can be moved elsewhere. As is true in most other cases where a volume is performing poorly, the only way of decreasing the impact of the remaining metadata is to increase the number of columns in the plex (stripe it across more disks).

The simplest approach to increasing the number of columns is to attach a new plex that contains more columns (disks) than the original plex, then remove the original plex(es). If you are running VxVM 3.0 or above, you can also use online relayout, which can get you to the same state without requiring quite as much disk space along the way. Both of these methods can be accomplished with vxassist.

Backups Are Hard, Restores Are Even Harder

The only practical way to back up a filesystem with a large number of files is via the raw device, since backing up each individual file sequentially would take a very long time. However, it's critical that you have a consistent version of the filesystem to work with for the duration of a backup. Snapshots, either at the volume or filesystem level, are generally the best approach. The FlashBackup extension to NetBackup provides an easy way of automating backups via filesystem snapshots. It also has the advantage of working with both VxFS and UFS filesystems.

Restores of millions of files also need to be done through the raw device, if at all possible, since they

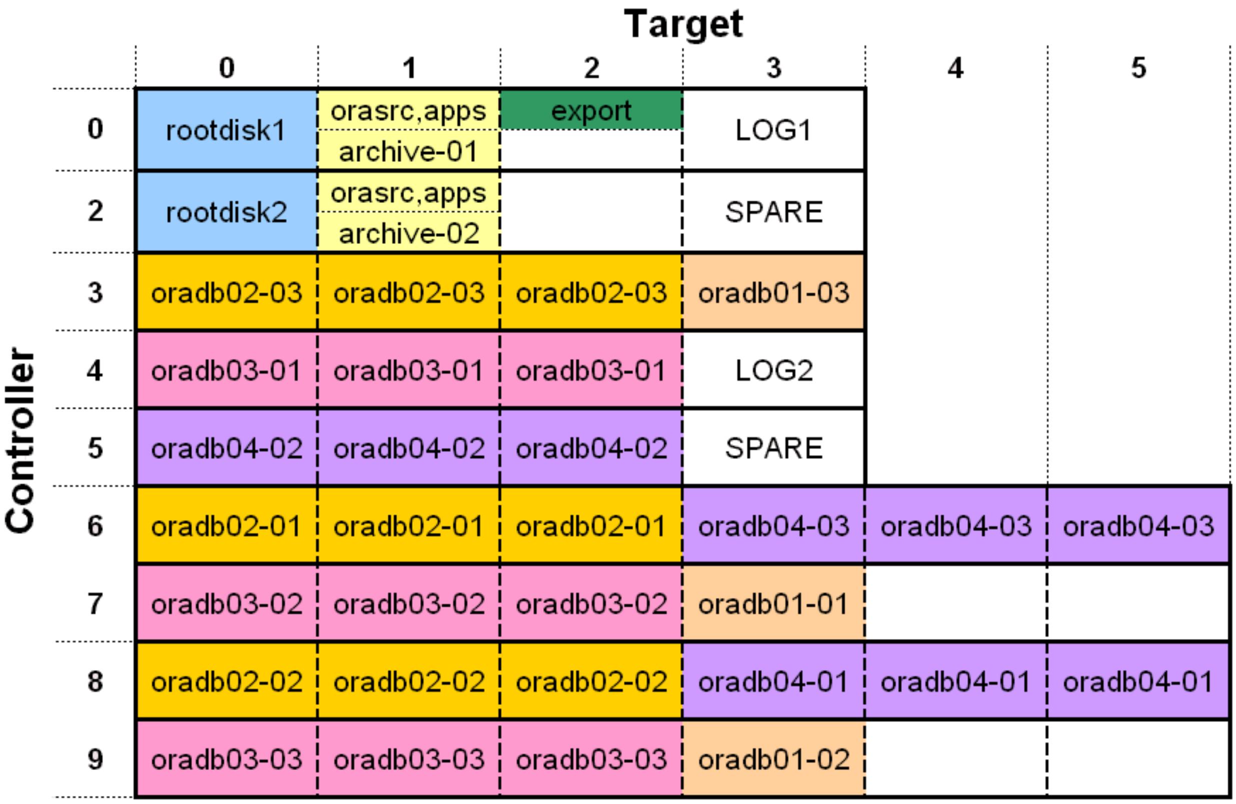

Figure 3: Example disk re-layout plan.

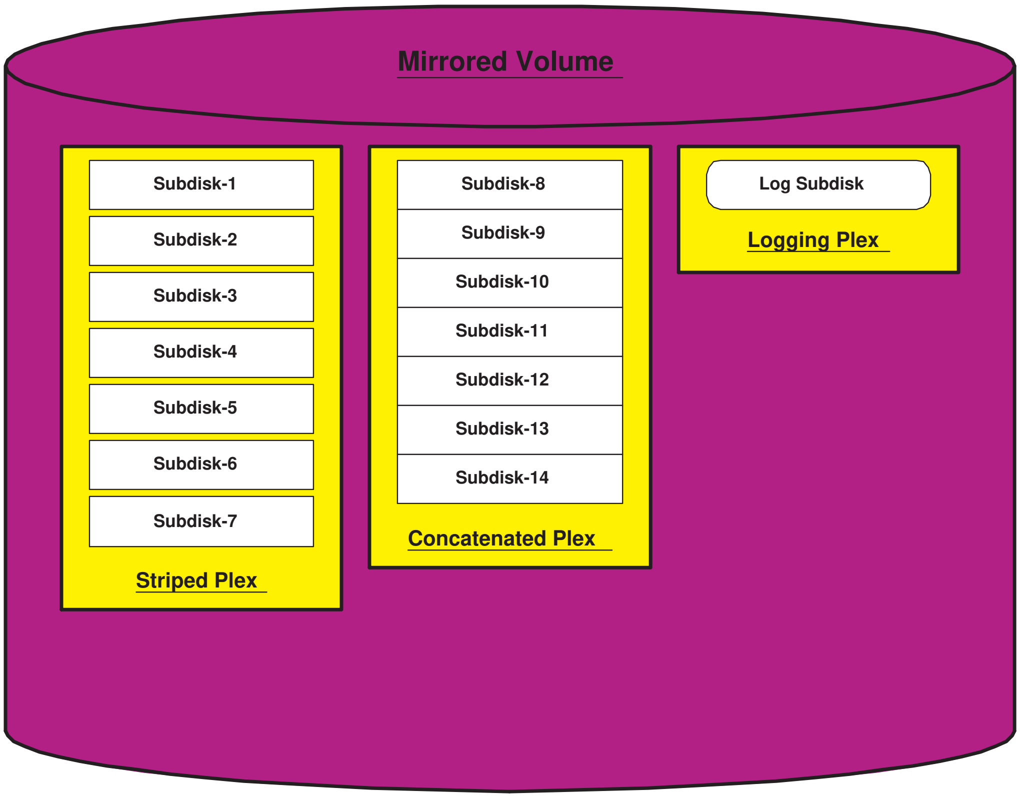

Figure 4: Example of mirrored volume with striped, concatenated, and logging plexes.

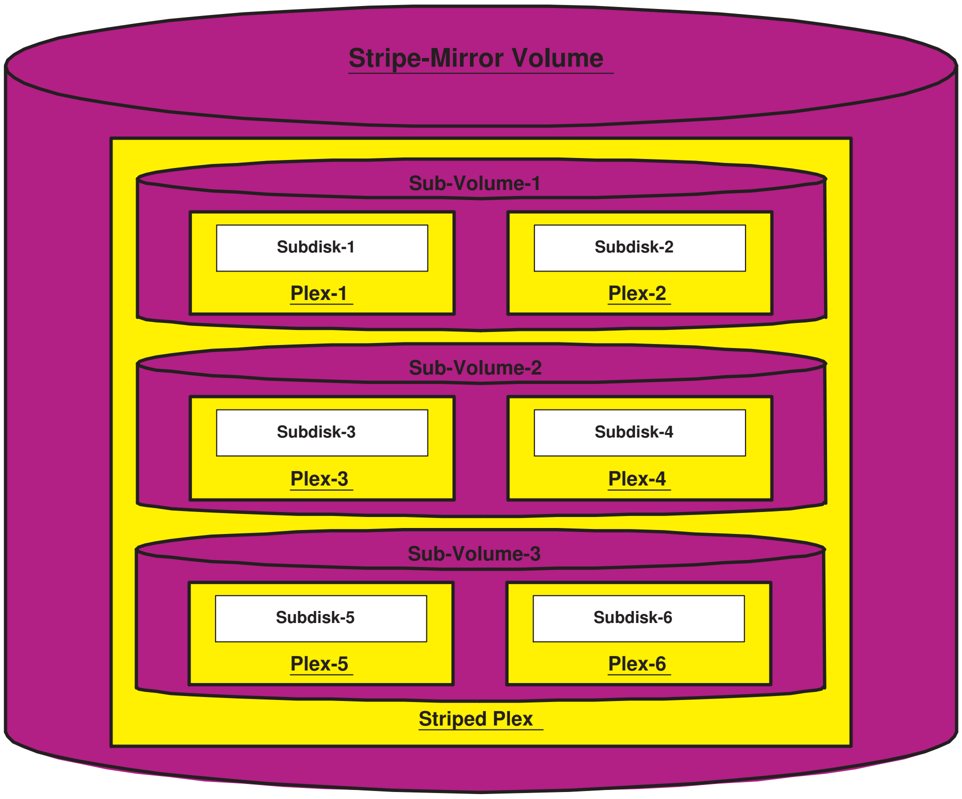

Figure 5: Example of layered volume with sub-volumes. This is also known as a striped-pro volume.

experience similar performance problems to backups. Even though incremental backups are possible and relatively fast with FlashBackup, incremental restores are another matter. Incremental restores are done through the filesystem instead of through the raw device, as other restore utilities like vxrestore and ufsrestore do. This can literally take weeks to complete when you have tens of millions of files. If you have that many files, you're probably better off only doing full backups, since they restore directly to the raw device. We've even seen full restores with FlashBackup take less time than the original backup since the restore doesn't require the inode map.

High Transaction Rate Systems

Extremely Busy Disks

From the info provided by vxstat or other tools, you'll probably find that certain disks are being used in too many "hot" volumes, or are otherwise unable to keep up with the I/O activity. If there are only a few busy volumes on the system, it's often possible to have only a portion of each physical disk in a busy volume and use the rest in more idle volumes. This allows you to increase the number of columns without having to add a lot of extra storage.

However, if almost every volume is extremely busy, you may not be able to fully utilize your disks. This is often a problem on systems with large disks, where the number of transactions per second is more than these disks can possibly keep up with. When possible, try to use small, high performance disks for your busiest volumes. If you're stuck with larger drives, consider preallocating up to 50% of each drive for a scratch volume, ensuring that other administrators don't accidentally over-commit them.

General relayout

The importance of carefully planning storage layouts before a system goes live can't be stressed enough. However, even after a system has been deployed, it's often possible to reorganize things. Start by creating a storage layout grid, similar to Figure 3.

Even if you only have one free disk available, it should be possible to gradually move data until it looks like your ideal storage plan. Online relayout may be your only choice if you don't have many free disks. It should be possible to make all of these changes without any service interruption, although the process of moving data can have a serious performance impact.

Glossary

Subdisk – A consecutive portion of a physical disk analogous to a partition. One or more subdisks can be grouped to form a plex.

Column – Groups of one or more subdisks that constitute a single stripe within a plex. For example, a 4-way striped volume would have four columns.

Plex – Typically, a set of subdisks that contain a complete copy of the data on a volume. Logging plexes are another type of plex which stores Dirty Region Logging data.

Volume – A collection of one or more plexes which make up a logical disk. It can hold a filesystem, swap device, or a raw device for a database.

Sub-volume – Starting with VxVM 3.x, you have the ability to create layered volumes where, for example, you can stripe across a set of mirrored subvolumes.

References

VERITAS File System System Administrator's Guide Release 3.3.2, VERITAS Software Corporation, May 1999.

VERITAS Volume Manager 3.1 Administrator's Guide, VERITAS Software Corporation, July 2000.

VERITAS NetBackup FlashBackup 3.4 System Administrator's Guide Unix, VERITAS Software Corporation, June 2000.

VERITAS User mailing lists, maintained by Doug Hughes: http://mailman.eng.auburn.edu/mailman/listinfo/ .

Repository of helpful storage management hints, maintained by Doug Hughes: http://www.eng.auburn.edu/pub/mail-lists/ssastuff/ .

Appendix: Third-party tools

Cricket

This tool is written and maintained by Jeff Allen. It can be downloaded from http: //cricket.sourceforge.com .

emcprint

The emcprint script produces It prints the director controller LUN for each disk that has an EMC name. This assumes you have run the createMatrix script (depends on symdev, included with EMC's symcli tools, a separately licensed product). This and the following three utilities are available from http://www.vxideas.org.

save-vxlayout

This tools preserves critical Volume Manager layout information and gives you a fighting chance of recovery if your configuration database gets corrupted. This sort of corruption can be caused by a split-brain scenario, where two systems try to update a disk group at the same time. It can either periodically write this backup info locally (to a file on a non-shared disk group) or email it off of the machine. We've also found it very useful for remote troubleshooting to use this to collect all of this info on a central management machine.

vxstat2gnuplot

This tool generates graphs of performance statistics of various VxVM-managed objects. This also depends on running rotatevxstat (which is also useful for determining bandwidth requirements for SRVM).

drive-status

This is a simple script which, when run from cron, notifies you of downed tape drives in NetBackup. This seems like something that NetBackup should provide natively, but at least as of version 3.2, it missed this. In large NetBackup environments, downed tape drives are often the leading cause of failed backups.

Author Information

Karl Larson attended Harvey Mudd College, majoring in Biology. He accepted a job as a Systems Engineer for WebTV Networks in 1996, where he helped support their production environment and scale it to support nearly a million users. He left to work at GNAC starting in 1999, where he was a Systems Architect, among other roles, helping design highly available customer environments. He has recently started work at Tellme Networks as a Senior Systems Engineer.

Todd Stansell attended UC Davis, majoring in Computer Science Engineering. He has been an employee of GNAC since early 1998, supporting customers such as WebTV, Exelixis, and Redherring. At GNAC, both Todd and Karl have worked with VERITAS Consulting to support some of their most critical customers.