

STORAGE SWITZERLAND

COST EFFECTIVELY SCALING STORAGE PERFORMANCE



George Crump, Senior Analyst

Few storage managers, at least those with budgets to adhere to, look forward to tackling a storage performance problem. Everyone knows it is going to be expensive and the instinctive approach of throwing more hardware at the problem by using faster drive technology does not always deliver the needed performance boost. Traditional dual controller storage architectures, and even the more modern 'scale-out' systems can't typically do the job. Cost effectively scaling storage performance is going to require a different storage architecture, one that can scale both performance and capacity in a separate and efficient manner.

When dealing with a performance problem most storage managers focus on the drive technology itself. However, a storage system has many components, the storage controller, the storage software, the I/O ports to the storage network, the I/O ports to the storage drive shelves as well as the drives themselves. And, drive speed is only part of the problem. In fact, a storage controller with the right level of innovation may be able to generate significantly better performance from the same number and type of drives than an inefficient system. Each

component in the storage system has to be optimized to deliver maximum performance from the rest of the system.

The Problems with Traditional Architectures

When trying to increase the storage performance on a traditional, dual controller architecture there are three weak links that must be understood. As stated earlier the drive technology is the first place that most storage managers look, and unfortunately it's often the last. Drives can be made to deliver data to the storage controller as fast as the shelf can handle it. The problem is solutions that just improve drive performance are expensive and should be used sparingly. These solutions typically include using a faster RPM drive to reduce seek time, short stroking those drives so that just the fastest outer tracks are used or adding solid state storage so there are no concerns about drive rotational speeds at all. Each of these solutions becomes increasingly expensive and while they are viable, should be a last resort because of that expense.

The other problem with focusing on drive components is that the maximum performance of those drives may never been reached because of bottlenecks elsewhere in the storage system. It may make more sense and be more cost effective to start at the top of the storage 'stack', the storage controllers, and work down to the drive layer. With this process when the storage manager gets to the point that they need to upgrade the drive technology they will know they're getting all the available performance from those drives. And, in most cases, they will need less of the higher cost drive technology.

The first performance pain point that should be explored is the storage controller itself. The speed of this controller and the amount of work that it can perform are critical to maintaining high performance throughout the rest of the storage architecture. In today's high performance database and virtualized server environments these controllers can quickly get flooded with server I/O requests and, depending on the drive type and number of drives, can also be overwhelmed by the storage shelves.

The standard for most storage systems has been a dual controller architecture. These controllers are sized to handle a specific number of IOPS and a specific number of drives. If purchased with only today's demands in mind they are the model of efficiency, and they generally don't take up much space since there are only one or two controllers per set of storage shelves. The controller CPU, I/O ports and drives could typically be run at near capacity. The problem is that storage shouldn't be bought to just meet today's demand, it has to be able to handle ever increasing requirements for IOPS and capacity. It needs to scale, and many legacy, dual controller architectures struggle to provide a scalable option.

In dual controller architectures, once the flooded state occurs the only solution for the storage manager is to upgrade to the next fastest storage controller or buy and manage multiple storage systems based on that same controller. This is of course an expensive, time consuming and potentially downtime sustaining event. The dual controller architecture also leads to buying more

processing power well in advance, which means paying extra for storage performance that may not be used for years to come.

Scale-out storage was supposed to fix the problem of not being able to add controller processing power and eliminate this requirement to pre-buy storage performance. Customers are instead encouraged to 'pay as you grow'. But scale-out storage also has a problem, it wastes storage controller resources. Scale-out storage systems typically add capacity, bandwidth and processing power *in unison*. This is not how storage performance problems typically occur. Environments either have a capacity problem, a bandwidth problem or an IOPS problem, not all three at the same time. As a result the scale-out model is often left with too much of one thing and not enough of the others, and as the environment scales the gap becomes worse. The reality of the pay as you grow model is that the storage manager is paying to grow three components when in most cases only one component needs to. Scale-out storage was a good start but there needs to be a finer level of granularity on what actually "scales".

[Storage-at-scale solutions](#) like those offered by [BlueArc](#) are another approach. They take the initial efficiency of a dual controller system, add the capabilities of scale and combine it with intelligent software and hardware for maximum utilization of all the resources.

The first key component in a storage-at-scale solution is to derive maximum performance from each storage controller, which in the BlueArc case is a NAS head. To do this requires intelligence. For example most storage controllers are overwhelmed by metadata operations (information about the data they store) before they are overwhelmed by other tasks. This means in file rich environments or those with large-sized files like media or virtual server images, larger controllers, more drives and eventually extra storage systems are added prematurely. If metadata is specifically focused on, the per controller performance can be accelerated and less of the above components are needed, leading to a significant cost savings.

Another example is to provide caching for read operations. With solid state storage becoming affordable, large caching tiers can be created very cost effectively. Since most environments are read-heavy, moving read intensive data to a cache tier can significantly increase performance. It's important to realize that if read operations are accelerated this leaves more potential storage controller resources to deal with write operations. In short, accelerating reads improves write performance.

By maximizing all available storage resources the time and frequency of a storage system upgrade can be delayed significantly. At some point though, even the best designed storage system will reach the maximum capabilities of its controllers. Once this has occurred then the storage systems must be able to provide a near linear performance upgrade by clustering multiple systems together. The cluster needs to be active/active and operating on the same storage network. The goal is to increase performance without increasing storage management time.

There is also a 'right sizing' objective to look for. The dual controller systems of the past could certainly be attached to the same storage network but had to be managed individually, increasing administration time. The 'modern' scale-out architectures encourage adding more storage nodes, essentially more storage systems, which may provide the scale needed, but sacrifice efficiency. There is also a latency issue to be concerned with in scale-out storage systems. As the node count reaches double digits the intra-node communication could become a bottleneck.

In any storage architecture adding nodes means adding costs but so does replacing storage systems because the current one has reached full capacity. Cost effectively scaling storage performance is accomplished by getting the maximum performance out of every resource that a storage system has. And it means adding intelligence to handle specific operations that can cause resource imbalances. By maximizing individual storage controller resources, storage-at-scale systems can limit the amount of additional nodes that need to be in the cluster and satisfy almost all performance demands with single-digit node counts. Fewer, more efficient nodes is the key to cost effectively scaling storage performance.

About Storage Switzerland

Storage Switzerland is an analyst firm focused on the virtualization and storage marketplaces. For more information please visit our web site: <http://www.storage-switzerland.com>

Copyright © 2011 Storage Switzerland, Inc. - All rights reserved