

Thin provisioning goes mainstream

By Ray Lucchesi

There are many benefits to be gained from thin provisioning, but there are also some potential drawbacks.



With the recent spate of announcements from a variety of vendors regarding thin provisioning, it is clear that this technology is rapidly expanding into the IT mainstream. NAS devices have long supported thin provisioning, but only recently has the feature moved into the larger block-level storage subsystems space.

What is it?

To use an analogy, my children occasionally play in rooms with inflatable castles and other structures. Thin provisioning is like an empty hall with a number of inflatable castles. The castles cannot inflate bigger than the hall, and each castle has its own inflation unit. In this analogy, the castles are LUNs, expanding (or inflating) whenever new block addresses are written to them. The size of the hall is either installed storage capacity or the storage defined to a thin-provisioned partition. In contrast, hard allocation is like building a house where each room is a LUN. You need to know the size of each room of the house ahead of time, and once the house is constructed, the only way to alter the size of a room is to demolish some nearby rooms.

Thin provisioning provides an alternative to the traditional method of hard allocation of storage for LUNs. Instead, thin provisioning allows allocation of storage space to applications or users in excess of actual physically installed storage. That is, a group of applications or users may be allocated space, which in total exceeds the available installed physical space. As a result, more applications or users can be

assigned to a particular storage subsystem than would be allowed under the hard allocation methodology. Most thin-provisioning products operate similarly to the familiar operating systems and/or virtual memory systems that use page tables to convert virtual memory addresses to physical memory addresses. Specifically, thin provisioning creates a page table to convert virtual LUN block addresses to physical LUN block addresses in real time, transparently to applications and users.

What are the benefits?

The overall benefit of thin provisioning—minimizing actual physical storage needed—is multi-faceted. For example, the need to purchase additional storage can be significantly delayed, resulting in current capital expense savings. Furthermore, savings can also be significant due to ongoing storage price erosion. Even more basic is the fact that fewer spinning disks reduce power and cooling costs as well.

Thus, cost savings from thin provisioning can be significant. However, the benefits don't stop there. Thin provisioning can positively affect operating efficiency of a storage subsystem. Again, the benefits are intertwined and not entirely obvious. Perhaps most important when addressing operating efficiency is thin provisioning's ability to utilize all available storage with many applications or users. Thus, assuming available physical storage, thin provisioning allows additional new data to be written and stored without administrator activity and without application or user interruption. Before thin provisioning, operations would have been negatively impacted to

change storage allocation both by interrupting the application and by the need for prompt administrator intervention.

Storage administrators are further aided by thin provisioning in that estimating application or user storage requirements is no longer necessary. All LUN storage can be configured to the maximum LUN size allowed, disregarding—at least during configuration—the finite physical storage in the subsystem.

Another area that has both storage administrators and their CIOs concerned, and where thin provisioning provides a significant advantage, is the often-abysmal storage utilization of non-thinly provisioned LUNs. Over-provisioned LUNs rarely exceed 50% storage utilization. In contrast, thinly provisioned LUNs are typically 100% utilized except in rare cases (see below).

Besides improving storage utilization, thin provisioning can improve application performance. For example, many thin-provisioning vendors automatically stripe LUN data across all disks in the subsystem or thin-provisioned partition. Without thin provisioning, the striping performance boost is achieved by RAID N+0 technology. However, this technology limits the number of disks available for a stripe by the number of RAID groups used. In contrast, thin provisioning is not likewise limited and can take advantage of all storage in the subsystem or thin-provisioned partition. As such, more disks are available to service I/O.

What's the downside?

As is usually the case with new technology, the benefits often give rise to increased com-

plexity by requiring more intelligence and logic. In the best case, end users, administrators, and CIOs would be impacted only minimally by the layer of complexity added by thin provisioning. However, this added complexity might result in functionality problems. Moreover, because of the global nature of thin provisioning these problems may negatively impact the entire storage subsystem.

The benefits of thin provisioning can also be decreased, or even erased, if a thinly provisioned subsystem is allowed to run out of space just as an application is attempting to write a new block. All new block writes are brought to a halt because the thinly provisioned subsystem has transparently allowed all of the subsystem capacity to be utilized. Non-thinly provisioned subsystems can also encounter storage deficiencies with hard-allocated LUNs. However, operations can be resumed more easily than thin-provisioned products by simply reconfiguring readily available, underutilized, hard-allocated LUNs.

As such, vendors of thin-provisioning products have provided for space warnings at designated intervals. However, these space warnings—although critical to the smooth operation of the subsystem—are elementary at best. A comprehensive plan for reacting to the space warning is paramount. A good plan should be based on historical capacity consumption over a specified time period (usually one week) and the time needed to order, deliver, and install additional storage. A simpler and less-comprehensive plan used by some administrators would hard-allocate a spare LUN and then delete that LUN for immediate additional storage when needed. Obviously, this latter plan would decrease the benefit of thin provisioning.

Will it work for you?

The decision to implement thin provisioning is not an easy one, as both the advantages and

disadvantages are like high-stakes poker: Gains or losses can be significant. However, consideration of some factors may make the overall decision more clear. These factors might include the following:

- **Willingness or ability to be operationally vigilant:** As discussed above, for thin provisioning, the lack of attention to space warnings can be disastrous. Thus, an unwillingness or inability to vigilantly monitor and react to space warnings may dictate your ultimate decision to forgo thin provisioning.
- **Current storage use behavior:** Applications, file systems or databases that always consume fresh storage space before re-using already written but available space are not good candidates for thin provisioning. A preponderance of this type of storage use behavior may again simplify the thin provisioning decision.

Once the thin-provisioning decision has been answered affirmatively, still other considerations, particularly as to vendor selection, become more important. Vendors differ dramatically in at least three areas in their approach to thin provisioning, including the following:

- **Storage page granularity:** Vendors provide widely different storage page granularity, ranging from kilobytes to gigabytes. At first glance it would seem advantageous to use the smallest granularity because it results in better storage utilization. However, such utilization comes at a cost. Specifically, smaller granularity requires significantly more control store or cache to hold the page table entries generated to describe a LUN. Small page granularity also may be a poor decision where the storage maintains a heavy sequential workload because disk subsystems can become highly fragmented, negatively impacting performance.
- **Random block writes:** Many thin-

provisioning vendors mitigate the devastating effects of applications that write to random block addresses by only allocating the page of storage needed to support the lone block being written. Some vendors, however, force the allocation of all the intervening storage pages to the LUN. Thus, the random writing application unnecessarily uses free space and can quickly fully populate a LUN. In this situation, the application may not be a good candidate for thin provisioning, but this can be better controlled by some vendors' implementations.

- **Thin-provisioning licensing:** Some vendors bundle thin provisioning into the storage hardware purchase price, while others charge for it separately on a capacity-used basis. Still other vendors charge based on the total capacity dedicated to thin-provisioned storage, whether or not it is used.

Conclusion

Thin provisioning, long available to NAS users, has only recently moved into the block-level storage subsystem arena. The cost, performance, and utilization benefits of thin provisioning can be substantial in this space. However, unwillingness or inability to be operationally vigilant can be deadly to thinly provisioned subsystems. In addition, a particular vendor's approach to thin provisioning can be very important. But, given appropriate vigilance and a proper determination of current storage use behavior, thin provisioning can provide a superior alternative to the traditional method of hard allocation. □

Ray Lucchesi is president of Silverton Consulting (www.SilvertonConsulting.com), a storage, strategy, and systems consultancy

© 2007 InfoStor

About the author:

Ray Lucchesi is president of Silverton Consulting, a Strategy, Storage & Systems consulting services company, based in the USA offering products and services to the data storage community.

<mailto:info@silvertonconsulting.com>

<http://www.silvertonconsulting.com>