

Sybase Adaptive Server® Enterprise 15 Very Large Storage Systems

BY JEFFREY GARBUS, SOARING EAGLE CONSULTING

From its inception, Adaptive Server Enterprise (ASE) has been a high-volume Online Transaction Processing (OLTP) or Decision Support System (DSS) or mixed-use database, capable of storing, managing, and retrieving an enormous amount of data.

Over time, the definition of the word “enormous” has changed.

In the early 1990s, we designed and rolled out an application which contained about 65 gigabytes of data. At the time, our contemporaries wondered if the system would even accommodate a table with 2.5 billion rows of data in a single table (it did). It was, at the time, the 4th largest ASE installation in the world. Anything over 20 gig was considered a Very Large Database (VLDB). Within a few months it had doubled in size, and a few months later doubled again to 250 gig, a monster in those days. The entire project was cost-justified in storage savings: a 2 gigabyte DASD device (for their DB2 applications) cost approximately \$40K, while a 2 gigabyte SCSI drive cost about \$2K.

Today, you can buy an external 1Terabyte hard drive for under \$100. Times change.

In this past year, I have had multiple customers accelerate their migrations to ASE 15 with one goal in mind: taking advantage of Very Large Storage Structures (VLSS) in ASE 15 so that they can increase the size of their growing, 4 terabyte databases to something with an almost unlimited upside (yes, famous last words, but with a potential upside of 2 billion devices of 4 terabytes each, I'll risk eating those words).

The idea behind VLSS is that Sybase has modified the ASE system tables to accommodate the larger limitations.

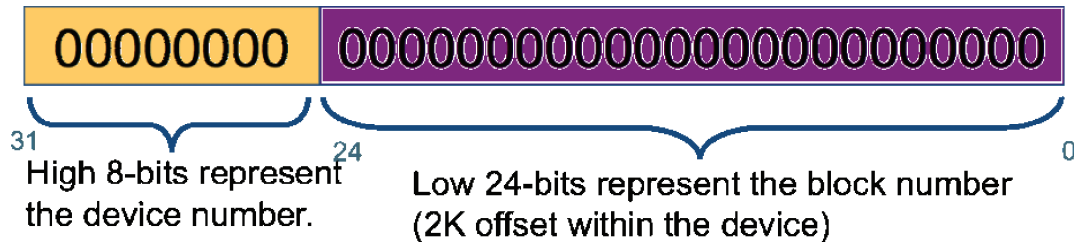
In this article we're going to discuss old structure, new structure, and practical limitations.

PRE-ASE 15 STORAGE

Prior to ASE 15.0, ASE was limited to 256 logical devices and 32GB for individual devices. Note that when these limits were originally put in place, 20 gig was thought to be a Very Large Database (VLDB), and it was years after before anyone attempted one. In addition, because of an intra-database addressing limitation, an individual database was limited to 4T (if you have a 2K page size) and 8T (if your page size is larger). These limitations were irrespective of physical device or OS-level limitations.

This was driven by the fact that ASE's storage was organized by virtual pages using a 4 byte bit mask. The virtual page was the mechanism the server used to connect the logical concept of a page with the physical disk. The high order byte of the vdevno column in the sysdevices table was used to encode the device id, and consequently the byte of 8 bits limited ASE to 256 devices (8 bits as a signed integer). (Of course, with one device set aside for master, and perhaps others set aside for tempdb, sysaudits, etc., there were sometimes fewer than 256 available for data). The remaining 3 bytes were used to track the actual virtual page numbers—which considering a 2K page (the storage factor in sysdevices) provided a limit of 32GB per device. Note that theoretically, a 16K server could have large devices, but due the default of 2K and the implementation of sysdevices, a limit of 32GB was imposed.

Here's a picture of the vdevno column:



Additionally, each database had a limit of 8TB—this was based on 256 devices of 32GB each. But, one database may only have up to 2^{31} (2,147,483,648) logical pages, so its maximum size also depends on its logical page size:

2K page server	4TB
4K page server	8TB
8K page server	16TB
16K page server	32TB

Previous limits in early 10 & 11.x timeframes of the numbers of devices per database and device fragments limits were removed a long time ago (they had formerly limited a database to 32 devices. Without the device limit of 32GB, the max database size would be 32TB based on ~2 billion 16K pages.

These limitations caused consternation in a variety of shops with a real need to store database which, at least for today, are still considered very big.

VERY LARGE STORAGE SYSTEM IN ASE 15

In ASE 15, Sybase has implemented what is being called the “Very Large Storage System (VLSS).”

The vdevno/vpageno combined 4-byte integer from the “low” column has been split into separate columns. This has required the rewrite of a significant part of server code; page addresses impact the page number and page header of every physical page—not just the system tables. As a result, ASE now supports ~2 billion devices of 4TB each. It still has the 32,767 limit for the number of databases per server, with a practical limit of roughly 100 due to backup timeframes, etc.

As a result for databases, the upper limit is now driven by the integer representation for pageno and the page size or 32TB as described earlier. The theoretical server size is now 8 million terabytes (8 Exabytes) which unfortunately is limited by the number of databases to 1 Exabyte.

If you have an OLTP in the terabyte range, this is for you (but note the comments later on partions!). But, if you have a DSS system, you should be giving strong consideration to Sybase IQ.

So:

Devices	2^{31} (2 Billion)
Maximum device size	4TB
Databases/server	32,767

Yields Maximum Storage:

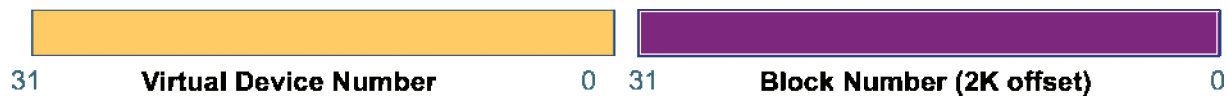
Database size	2^{31} pages * 16KB pg = 32TB
Theoretical DB storage	32,767 DB's * 32TB = 1 EB (exabyte) = 1,048,544TB
Theoretical server size	2^{31} devices * 4TB size = 8,589,934,592TB

Comparing pre-ASE 15 limits to newer (sic) limits, we get

Attribute	Old (12.5.x) Limit	New (15.0) Limit
Number of devices	256	2,147,483,648 (2^{31})
Maximum device size	32GB	4TB
	$32GB <- 2^{24} * 2K \text{ vpg}$	
Maximum database size (2K / 4K / 8K / 16K)	4TB / ~8TB / ~ 8TB / ~8TB	4TB / 8TB / 16TB / 32TB
	$8TB <- 256 \text{ devices} * 32GB$	

Not bad; pre-15, an entire database/server could only take up 4TB; with ASE 15, we can have 2 billion devices of that size.

In ASE 15.0, the virtual page number is represented by two 32-bit values. One is the device number; the other is the block number



A WORD ON PARTITIONS

Storing the data is half the battle. Maybe a third.

The rest of the battle involves data manipulation and maintenance.

When you insert a row, how does the server know where the next empty page is? When you retrieve a row, how long will a table scan take? How do you backup a 10TB database? Run dbcc? Update your statistics?

Databases in the terabyte range are nontrivial to manage at this point in history. (In another decade, they'll fit in my wristwatch). For today, we need to plan a bit.

With the advent of ASE 15, Sybase has added real semantic partition management to Adaptive Server Enterprise. This means that you can separate your data into a variety of partitioning schemes (subject of a separate white paper).

Partitioning, though, is extremely relevant to large tables (which tend to go along with large databases), because of everything from insertion rates to data archival.

Partitioning lets you:

- Partitioned indexes, which yields faster access performance, smaller index structures, and improved parallel search
- Partitioned data, which allows you to organize the data based upon your own rules (hash, data range, list partitioning, round-robin partitioning), which lets you either spread io across multiple LUNs, or to allow archiving to hit only one of several LUNs, without interfering with online access
- Partition-aware maintenance, which lets you (for example) update statistics or run dbcc on only the more recent partitions. You can also truncate, reorg, or bcp on a partition basis

SUMMARY

Sybase, with the release of Adaptive Server Enterprise 15, has effectively removed size limitations from databases and servers. In addition, the enhancements have provided enhanced access techniques not only for data access performance, but to enable management of the larger databases.

Jeff Garbus has 20 years of expertise in architecture, tuning and administration of Sybase ASE, Oracle, and Microsoft SQL Server databases with an emphasis on assisting clients in migrating from existing systems to pilot and enterprise projects. He has co-authored 15 books and has published dozens of articles on the subject. Mr. Garbus is the CEO of Soaring Eagle Consulting, an organization that specializes in assisting businesses maximize database performance www.soaringeagle.biz.