



Craig S. Mullins

[Return to Home Page](#)

August / September 2004



zData Perspectives

by Craig S. Mullins

What is Large?

Every now and then some sage consultant will offer advice like “Large DB2 table spaces should be partitioned” or “Bind your DB2 applications using ACQUIRE(ALLOCATE) and RELEASE(DEALLOCATE) for high volume transaction workloads.” But how useful is this advice? What do they mean by large and high volume? Terms such as these are nebulous and ever changing. Just what is a **large** database today?

Of course, the standard answer of “it depends” applies – it depends on your site, your mixture of data types, and your particular requirements. But is that any more useful? Let’s try to define the term and put some hard numbers around it.

First of all, the question begs to be asked: “Large in terms of what?” The first thing that springs to mind is the actual size of the database. But are we talking about a single table space or an entire database? And sometimes we are even referring to the size of an entire subsystem (such as for SAP R/3 implementations).

If we are at the table space level, what is the granularity of measurement for determining what is large? Do we talk in terms of number of rows or number of pages? Or just the amount of disk space consumed? And do we count just the base data or add up the space used by indexes on that data as well?

And what about the type of data? Is a 20 GB database consisting solely of traditional data (that is, numbers and characters; dates and times) bigger than a 50 GB database that contains non-traditional BLOBs and CLOBs? From a purely physical perspective the answer is obvious, but from a management perspective the answer is more nebulous. It may indeed be more difficult to administer the 20 GB database of traditional data than the 50 GB database of large objects because traditional data consists of more disparate attributes and is likely to change more frequently.

Another issue is just what are we counting when we say we have a large database? Do we count copied and denormalized data? And what about free space? There are two schools of thought: one says if it is in a database file, then it counts. Another says, let’s only count the core data. From the perspective of the DBA though, you have to count everything that needs to be managed – and doesn’t everything need to be managed?

One useful measure of large databases is offered by [Winter Corp.](#), a research and consulting firm specializing in database scalability. Winter Corp. applies its research and analysis resources to measure industry database implementations in terms of size, rows/records and workload. Its most recent reports outlined the following databases as the largest:

In terms of actual size, Winter Corp. reported the largest database to consist of 828 TB of data. In terms of workload, 51,448 transactions per second (tps), was the highest figure. And the largest number of rows for a database was 496 billion rows. This data comes from Winter's 2003 study which was also the first to award winners for volume of normalized data. The grand Prizes for normalized data volume went to a 94.3 TB database. Indeed, these are all very large – and likely well beyond the scope of what a “normal” IT environment would call large.

So you need to be prepared with the criteria for what establishes database largeness at your shop. Is it a management issue? A planning issue? It better be both of those, but sometimes it is a braggadocio issue, too! You know, being able to say “My DB2 can beat up your DB2.”

Furthermore, the granularity of the object being discussed is important, too. Winter Corp. researches at the database level, but DBAs are probably more interested in managing at the table space level, at least for DB2. So what is a large table space? A good place to start is probably 4 GB. To specify a value greater than 4GB, you must be running DB2 with DFSMS V1.5 or greater, and the data sets for the table space must be associated with a DFSMS data class defined with extended format and extended addressability. DFSMS's extended

addressability function is necessary to create data sets larger than 4GB in size. Of course, depending on your shop and its requirements this might be too high... or even too low.

So here we are, near the end of this column and we seem to have more questions than answers. So how about some advice? First of all, when determining what is a large table space for your shop, do it in terms of the number of pages, not the number of rows. You can use this number to easily compare the size of one table space to another, whereas you cannot if using number of rows because row size can vary dramatically from table to table. And count everything that is being persistently stored in the database: data, indexes, free space, etc. If it is being stored it must be managed, and therefore impacts TCO. Stripping out everything but normalized data only matters when you are worrying about who has the biggest database, and we should be more worried about assuring the availability and manageability of our big databases!

The Bottom Line

One thing can be said for sure, though – and that is this: our databases are getting bigger. The latest Winter Corp. report verifies this fact as it shows that the size of world's largest databases more than doubled since 2001. We will soon be talking more frequently in terms of terabytes instead of gigabytes, and then petabytes after that. Who said life as a DBA was boring?

From [zJournal](#), August / September 2004

.

© 2004 Craig S. Mullins, All rights reserved.

[Home](#).