Craig S. Mullins

Return to Home Page

June / July 2006

The Resource for Users of IBM zSeries & S/390 Systems
z/JOURNAL

zData Perspectives

by Craig S. Mullins

The Basics of DB2 Buffering

One of the most important areas for tuning DB2 subsystem performance is memory usage. DB2 for z/OS uses memory for buffer pools, the EDM pool, RID pool and sort pools to cache data and structures in memory. The better memory is allocated to these structures, the better DB2 will perform.

When allocating DB2 buffer pools, keep these rules of thumb in mind:

- Don't allocate everything to a single buffer pool (e.g., BP0); use a multiple buffer pool strategy.
- Explicitly specify a buffer pool for every table space and index.
- Isolate the DB2 Catalog in BP0 (and BP8K0 as of V8); put user and application DB2 objects into other buffer pools.
- Consider separating indexes from table spaces with each in their own dedicated buffer pools.
- Consider isolating heavily hit data into its own buffer pool to better control performance.
- Consider isolating sorts into a single buffer pool and tuning it for mostly sequential access (e.g. BP7).
- Consider separating DB2 objects into separate buffer pools that have been configured for sequential verses random access.

Forget about trying to follow a cookie-cutter approach to buffer pool management. Every shop must create and optimize a buffer pool strategy for its own data and application mix. DB2 offers several buffer pool tuning "knobs" that can be used to configure virtual buffer pools to the type of processing they support. The following parameters all can be changed using the ALTER BUFFERPOOL command:

DWQT –this value is the deferred write threshold; it is expressed as a percentage of the virtual buffer pool that might be occupied by unavailable pages. When this threshold is reached DB2 will start to schedule write I/Os to externalize data. The default is 50%, which is likely to be too high for most shops.

VDWQT – this value is the vertical deferred write threshold; it is basically the same as DWQT, but for individual data sets. The default is 10%, which once again is quite likely to be too high for many shops.

VPSEQT – this value is the sequential steal threshold; it is a expressed as a percentage of the virtual buffer pool that can be occupied by sequentially accessed pages. Tune buffer pools for sequential access (such as scans and sorting) by modifying VPSEQT to a larger value. The default is 80%.

VPPSEQT – this value is the sequential steal threshold for parallel operations; the default value is 50%.

VPXPSEQT – this value is assisting parallel sequential threshold; it is basically the VPPSEQT for operations from another DB2 subsystem in the data sharing group.

Prior to DB2 V8, hiperpools can be created to back up DB2 virtual buffer pools with additional memory. DB2 provides several tuning knobs for hiperpools, too, including HPSIZE to adjust the size of hiperpools and HPSEQT to adjust the hiperpool sequential steal threshold. Hiperpools are obsolete as of V8 though, so if you don't use them today you should probably spend your time migrating to V8 instead of implementing soon-to-be-obsolete hiperpools.

With the advent of DB2 V8, there is more memory at your disposal for DB2's use. V8 is able to surmount the limitation of 2GB real storage that was imposed due to S/390's 31-bit addressing. Theoretically, with 64-bit addressing DB2 could have up to 16 exabytes of virtual storage addressability to be used by a single DB2 address space. In terms of buffer pools, DB2 can support up to 1 TB for buffer pools. But keep in mind that this is a theoretical maximum. The actual maximum size of a buffer pool is still constrained by the amount of real storage available on you particular hardware. For example, the z990 mainframe can support only up to 256 GB of memory.

In addition to buffer pools, DB2 uses memory for the EDM pool. The EDM pool is used for caching internal structures used by DB2 programs. This includes DBDs, SKCTs, CTs, SKPTs, and PTs. It also includes the authorization cache for plans and packages, as well as the cache for dynamic SQL mini-plans.

With V8, DB2 breaks the EDM pool into separate pools: one for DBDs, one for the dynamic statement cache, and the final one for program elements (CTs, SKCTs, PTs, SKPTs).

As a general rule of thumb, shoot for an 80 percent hit rate with the EDM pool; this means that only one out every five times should a structure need to be loaded from disk into the EDM pool.

Finally, remember that buffer and EDM pool tuning are in-depth subjects that cannot be adequately covered in-depth in a single page column such as this. So, study those IBM DB2 manuals - and learn by doing. Additionally, there is much more to proper DB2 system performance tuning than memory tuning. Other system elements requiring attention include allied agent setup (CICS, TSO, etc.), network configuration, locking, logging, and Parallel Sysplex configuration and management for DB2 data-sharing shops.

From zJournal, June / July 2006

.

© 2006 Craig S. Mullins, All rights reserved.

Home.