



April / May 2011

zData Perspectives

Best Practices in DB2 Storage Management

by Craig S. Mullins

amazon

DB2 Developer's Guide: A...

\$53.99

Shop now

Although storage management can be an afterthought for the DBA it really shouldn't be. The cost of managing storage can be as much as 10 times higher than the initial cost of acquiring the storage. And the growth rate for disk storage was 37% between 1996 and 2007. So storage issues are vitally important and unless managed appropriately it can be very costly. Even so, it is common for storage-related issues to be relegated to the backburner by DBAs. But every DB2 professional should understand modern storage basics.

DB2 stores data in VSAM Linear Data Sets (LDS). Each table space and index space requires at least one, possibly more, VSAM data sets. But there are many other storage-related objects you will encounter with DB2 for z/OS, including Storage Groups (both DB2 and DFSMS), system data sets (logs, BSDS, etc.), image copy backups (stored on disk or tape), and others like DB2 library data sets and temporary data sets.

For one thing, the latest extent management features only work with SMS-managed data sets; if your data sets are user-managed then the old rules apply. For example, as of z/OS 1.7 system-managed data sets can have up to 123 extents on each of 59 volumes for a total of 7,257 extents. Otherwise the limit remains 255. Also, extent consolidation, introduced in z/OS 1.5, requires SMS-managed STOGROUPs. When a new extent is adjacent to old, they will be merged together automatically. This can result in some extents being larger than the PRIQTY or SECQTY specification.

Even if everything is SMS-controlled, extents can impact performance. Elapsed time can increase with multiple extents if there is heavy insert activity. For reads and updates the number of extents should not impact performance. Regardless, you no longer need to continuously monitor extents and clean them up immediately by reorganizing. It is still a good practice to periodically clean up extents, but there are other methods of reducing extents that are quicker and easier than REORG.

Storage-related information is available to DB2 folks from multiple sources including the DB2 Catalog, real-time statistics (RTS), RUNSTATS, STOSPACE, and so on, but the details are scattered all over the place and it can be difficult to gain a complete, accurate, and up-to-date picture. Furthermore, any historical view into DB2 storage usage has to be managed manually.

A responsible DBA with a comprehensive storage strategy will be able to ensure that all DB2 databases have sufficient allocation to satisfy business requirements. They will be able to answer questions like "Why is DB2 storage growing when our business is not?" Wasted storage will be minimized and a proactive approach to adding more storage when required will be adopted.

Although mainframe disk is usually equated to a 3380 or 3390, the actual physical disk is no longer as simple. Today's modern storage architecture uses disk arrays, or RAID (Redundant Array of Independent Disk). An array is the combination of two or more physical disk devices in a single logical device or multiple logical devices. The array is perceived by the system to be a single disk device.

Another relatively recent change is the use of IBM's DFSMS (Data Facility Storage Management System) to manage DB2 data. And as of DB2 9, DATACLAS, MGMTCLAS, and STORCLAS can be specified in DB2 Storage Groups. Using DFSMS with DB2 is the same thing to do these days because the new disk architectures, with concepts like log structured files and with gigabytes of cache, render conventional database design rules based on data set placement less relevant. In most cases, placement is not an issue, and when it is, SMS classes and ACS routines can be used. Keep in mind, too, that as of DB2 10, the system catalog must be managed by DFSMS.

Also, DB2 V8 introduced sliding scale secondary extents, which cause allocated extent sizes to gradually increase. This feature helps to reduce the number of extents for your DB2 objects as they grow in size over time. And it is useful when you do not know up-front how rapidly your data will grow.

In terms of storage best practices, it is a good idea to perform regular and proactive monitoring. Examples of things you should be tracking include:

1. space used by your entire DB2 system, individual DB2 databases, and of your table spaces and indexes;
2. monitoring the Storage Groups and the associated volumes of a DB2 system;
3. monitoring all VSAM data sets for all table spaces and indexes including Used, Allocated, Primary and Secondary Quantity, Extents and the Volumes they are on;
4. alerts for Page Sets of table spaces and indexes that reach their maximum size and maximum number of data sets;
5. tracking of image copy backup data sets, including HSM migration; and
6. a way to delete Image copy backup datasets that are no longer needed because of DROP, DROP/CREATE or MODIFY TABLESPACE.

Whenever possible, create alerts to automatically inform you of problems, shortages, and potential errors. Automate remediation tactics so that the alert tells you what happened, as well as what was done to correct the issue. Tools may be able to assist in automating reaction to shortages, potential errors, superfluous data sets, etc.

The better the database system works with the storage systems, the better your database applications will perform. And that is what it is all about, right?

What about extents? Many believe that modern storage devices render extent management obsolete, but that is not exactly true.

From [zJournal](#), April / May 2011.

© 2012 Craig S. Mullins,

DB2PORTAL.com

© 2021 Mullins Consulting, Inc. All Rights Reserved [Privacy Policy](#) [Contact Us](#)