



Craig S. Mullins

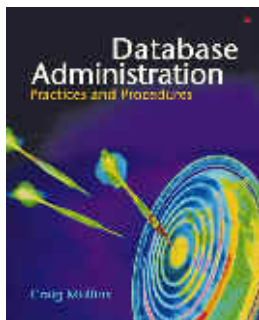
[Return to Home Page](#)

October 2007



The DBA Corner

by Craig S. Mullins



The Impact of Data Volume on Operational Databases

Operational databases are growing in size for many reasons. First of all, there is the overarching trend of more and more data being generated every year. Industry analysts peg compound annual growth rate for data at 125%. This means that we are more than doubling the amount of data we manage each year.

But things are actually more difficult than that statistic implies. This is so because of regulations and business requirements dictate that we must store that data for longer periods of time. So we are stuck managing more data for greater periods of time – and that can complicate matters as we attempt to wring performance and usability out of our databases.

Indeed, as data volumes expand, it impacts operational databases in two ways:

1. additional data stresses transaction processing by slowing things down, and;
2. database administration tasks are negatively impacted.

In terms of performance, the more data in the operational database, the less efficient transactions running against that database tend to be. This is so for many reasons. Table scans must reference more pages of data to return a result. Indexes grow in size to support larger data volumes, causing access by the index to degrade because there are more levels to traverse to return an answer. Such performance impacts are causing many companies to seek solutions that offload older data to archive data stores – more on that in a moment.

The other impact, database administration complexity, causes longer processing time and outages to perform traditional DBA tasks. Functions such as backups, unloads, reorganizations, recoveries, and disaster recoveries all take longer to perform as the database increases in size. And these tasks can cause outages, outages that expand as the database grows. In many cases the lengthened outages have become unacceptable. When you can no longer satisfy agreed upon service levels (e.g. time to recovery) you need to seek ways to lighten up the girth of your operational databases.

But as important as these operational performance and administration issues are, frequently they are ancillary to the regulatory issue of preserving authentic data over time. Although both are driving the need to move data from the operational

database into an archive data store, it is the legal requirements that are likely to have the biggest impact in terms of data volume expansion. Nonetheless, the operational issues are real and need to be managed.

A popular approach to handling this growing problem is database archiving. [Database Archiving](#) is the process of removing selected data records from operational databases that are not expected to be referenced again and storing them in an archive data store where they can be retrieved if needed.

By archiving data when it is no longer needed for operational purposes, administrators can relieve the pressure on their production databases while at the same time preserving the data that is no longer accessed by business transactions and reports. Because the data is preserved in the archive it can be easily retrieved if and when it is needed, for example in the e-discovery phase of a lawsuit or during an audit for compliance with governmental regulations (such as SOX or HIPAA to name but two of the hundreds of regulations that demand data retention).

When database archive policies are in place, older data that is no longer accessed is periodically archived out of the operational database. This has the effect of minimizing the impact of data growth, at least in terms of database and transaction performance. Yes, the archived data is still stored, so we are not likely to experience savings in disk storage, but the data is stored separately from the operational database and does not require the DBMS any longer. Archived data is separate and independent from the production systems from which it was moved and, therefore, business transactions and DBA tasks can be performed more efficiently and with minimized outages.

Furthermore, archiving dormant data from operational databases can reduce the threat of a data breach. Operational systems are more prone to attack than

archived ones. Not only are operational systems more visible, but once data is archived it typically is not accessible using the operational transactions and reports. By archiving database data you can reduce the temptation of internal breaches because the data is not as readily available, and reduce the possibility of external breaches because archived data typically is not exposed to large numbers of users over vast networks.

By instituting an archive policy for your burgeoning database data, overall productivity improves because user's requests are being resolved faster, downtime for administrative tasks is minimized, and new life is breathed into your operational systems. If your databases are growing to the point where their size impacts daily operations you should implement a database archiving solution to revitalize your operational environment.

From [Database Trends and Applications](#), October 2007.

© 2007 Craig S. Mullins, All rights reserved.

[Home](#).