Choosing a Capacity Management Solution

white paper

NEON
Enterprise Software, Inc.

Table of Contents

white paper

IMS Capacity Management and Performance

From its initial release, IMS customers faced issues of capacity management and database performance. As the volume of data grows, the need for more efficient space management and data access increases. With business data frequently reaching the terabyte milestone, capacity management and database performance have become critical issues for the life of many businesses.

Choosing a capacity management solution may be one of the most important decisions an IMS shop makes, potentially touching many areas of data management and business applications. In this article, we review the concepts and considerations that surround IMS capacity management and database performance. We then review two of the existing solutions that address this growing concern and discuss the pros and cons of each tool.

Why is Capacity Management a Problem?

Think about your first desktop computer. A 40-MB hard drive could run all of the applications that you needed, and store all of your documents and data with space to spare. Today, you can buy a PDA with six times that amount of space, and you would not even consider a desktop computer with less than a 40-GB hard drive. Both application size and the amount of data have grown beyond the capacity you imagined you could use even a few years ago.

Similarly, more than three decades of aggregating data pushes the limits of IMS and the applications that use IMS data. However, increasing IMS capacity involves more than just more DASD. The capacity limits of IMS are based on z/OS and IMS limitations, including:

- Maximum size of data sets (4 GB for VSAM or 8 GB for OSAM)
- Maximum of 10,000 data sets per IMS address space
- Maximum of 8,183 open full function database data sets (for DFSMS 1.4 and earlier releases)
- Availability of large databases during maintenance
- Longer run times for data access and recovery
- I/O contention for frequently accessed data

The impact of capacity problems on your business can be significant. If IMS exceeds available capacity, costly database downtime, recoveries, and unscheduled maintenance can explode your budgets, cause retrofits to applications, and ultimately impact your customers.

There are a number of solutions to capacity management and database availability issues. Some methods include database purging, segment compression, conversion to OSAM data sets, and the use of Fast Path databases to extend the capacity life of large databases. These methods are thoroughly discussed in, "Large Database Alternatives," by William Keene (available at www.neonesoft.com), and will not be explored here. Instead, this paper focuses on database partitioning solutions, which have both capacity management and performance benefits.

How Partitioning Solves the Problem

Partitioning spreads database records across multiple partitions in a manner that is transparent to database applications. Data set size limits apply to each partition, extending the capacity of the database as a whole. For example a database with 10 partitions will have a capacity of 40 GB for VSAM data sets and indexes, and 80 GB for OSAM data sets.

Partitioned databases also offer performance improvements by using one or more of the following methods:

- Parallel processing for routine database-management tasks, such as database reorganizations, and batch tasks.
- Partitioned indexes reduce data retrieval times.
- Reduced I/O contention for database resources.

white paper

The maximum number of data sets per IMS address space is not affected by partitioning. Following a strategy that uses more partitions with smaller data sets (for example 2 GB) will normally produce better database performance. However, in very large databases you may need to plan larger data sets to avoid the 10,000 data set/address space limit.

Partitioning Limitations

All partition solutions have certain restrictions. In general, partitioning tools require some changes to the database definition (DBD). Virtually paired logical relationships are not supported and must be changed to physically paired logical relationships. In addition, changes must be made to the way that data sets are dynamically allocated.

Considerations for Partitioning Solutions

After you make the decision to partition a database, you may want to evaluate your choice of partition solutions to determine the best options for your site. The following sections describe several areas that you should consider when you evaluate partitioning tools to use in your organization.

Ease of Implementation

When you consider paritioning tools, you want to know that the tool you install will fit well into your system, requiring few or no application changes. You should also consider the following questions:

- Does the tool you are evaluating require a great deal of initial setup or configuration?
- Can you implement partitioning in a staged manner, or will you need to form a strategy for all IMS before you convert to partitioned databases?
- How does the partitioning tool support special requirements, such as secondary indexes, logical relationships, and non-DBRC-managed databases?
- Can you return the partitioned database to an unpartitioned state if necessary?

Cost of Conversion

The cost of converting to a partitioned environment depends on more than just the cost of the tool. Look for hidden costs in the time involved if you have to redesign IMS to achieve partitioning. Also, look at the IMS and system resources required for the partitioned database.

For example, partitioning tools can create additional data sets that must be managed and maintained, and that can require more space that in an unpartitioned database. Additional log records can also be an issue.

The cost of training personnel to manage and maintain a partitioned database should also be factored into a partitioning project estimate. For example, if your personnel must learn DBRC to manage the new databases, then that cost may offset some of the partitioning advantages. Alternatively, if applications have to be changed to achieve partition independence, then the reduced maintenance time may be eaten up by application maintenance, at least in the short term.

Database Maintenance Issues

Overall, partitioning reduced the amount of time that it takes to maintain and manage an average IMS database. The solution that you choose should not interfere with your normal database maintenance process, allowing you to implement partitions without too many process changes. The tools we look at later in this document both have special database utilities to handle routine maintenance tasks, with some differences in approach.

Also, consider the need to support existing IMS features such as secondary indexes, logical relationships, particular database types, and performance requirements. In some cases, your site may have specific requirements that favor one solution over another.

HALDB and PDF Partitioning Solutions

The following sections compare the IBM HALDB partitioning solution to the NESI Partitioned Database Facility (PDF).

Conversion Process Comparison

The conversion of an unpartitioned database to PDF is very straightforward. Converting a HDAM or HIDAM database to PDF is much like adding data set groups. There are no changes necessary to primary or secondary indexes with PDF. There are no special steps or skills necessary when converting to PDF. The utilities can recommend key ranges when converting a HIDAM database to PDF to make conversion easier. Many customers have converted their first database to PDF in a matter of weeks if not days.

HALDB conversion is a completely new process. HALDB definition requires both DBD and DBRC changes. An ISPF-based Partition Definition Utility is provided, or some sites use DBRC INIT.PART commands to define HALDB partitions in DBRC.

Other conversion considerations include the following:

PDF	HALDB
Virtual pairing not supported, must convert virtual pairs to physical, logical relationships.	Virtual pairing not supported, must convert virtual pairs to physical, logical relationships.
DBRC supported, but not required.	DBRC required, so personnel must know DBRC. IBM suggests that application function testing not use HALDB to avoid the DBRC requirement.
Logical relationships can be to non-PDF databases.	Logical relationships must be to other HALDB databases.
Secondary indexes supported for PDF. Indexes can be partitioned.	Secondary indexes must be converted to PSINDEX and loaded separately.
No application changes required	Application changes are required to take advantage of partition independence and for applications that read secondary indexes (8-byte /SX field).

Database Structure and Planning Requirements

The types of supported databases, the number of data sets required, and supported database features can make a difference in the type of tool required to partition an IMS databse. The following comparison summarizes the database types and features supported by PDF and HALDB.

white paper

Support for	PDF	HALDB
Maximum data set size	4 GB for VSAM and KSDS data sets 8 GB for OSAM data sets	4 GB for both OSAM and VSAM data sets
Number of data sets per data set group	10	10
Maximum number of partitions and data set groups	127	1001
Secondary Indexes	YES	YES (PSINDEX)
Logical Relationships	YES	YES (only to other HALDBs)
HDAM databases	YES	YES (PHDAM)
HIDAM databases	YES	YES (PHIDAM)
Root-only HISAM databases	YES	NO
SHISAM databases	YES	NO
Virtual Paired Relationships	NO	NO
Index build	YES	NO
Index per partition	YES	YES
One index for all partitions	YES	NO

Index Support

PDF supports both recoverable and nonrecoverable primary and secondary indexes. Customers can restore the primary and secondary index using image copies and point-in-time recovery, or the index can also be rebuilt from the database. The HALDB primary index (PINDEX) and ILDS must be nonrecoverable and cannot be image copied by IMS. Any procedures using IMS image copies of primary indexes will have to be changed with HALDB. The ILDS must also be considered for HALDB image copy or point-in-time recovery processes

PDF secondary indexes are completely compatible with non-PDF, full-function secondary indexes and can be partitioned or nonpartitioned. The IDCAMS definition for a secondary index of a PDF database is the same as for non-PDF databases. The size of the secondary index does not increase when converting the target database to PDF.

HALDB requires that secondary indexes be converted to HALDB PSINDEXes when the database is converted to HALDB. The size of the PSINDEX record is increased by the 28-byte extended pointer set (EPS) and the size of the root key. The /SX field, which is part of the key, will also increase from 4 bytes to 8 bytes. The size of the HALDB secondary index increases dramatically for HALDB.

Logical Relationships

Both PDF and HALDB support logical relationships and require that virtually paired logical children be converted to physically paired relationships. The logical parent concatenated key (LPCK) must be physically stored with the logical child.

PDF uses symbolic pointing to locate the logical parent. Pointer resolution is not required after the logically related database is reorganized. HALDB uses the extended pointer set (EPS) and self-healing pointers to update the EPS after a logically related database is reorganized. See page 7 for more information about the pointer update process.

PDF allows a logically related database to be non-PDF. In other words, when a large database is logically related with several databases, only the large database needs to be converted to PDF.

HALDB requires that all logically related databases be converted to HALDB at the same time. HALDB does not allow a non-HALDB database to be logically related to a HALDB database.

Performance Comparison

Performance improvements are inherent in any IMS database partitioning solution because database access is spread across multiple data sets. Highly accessed large database performance can be improved by converting to either PDF or HALDB partitions.

Support for	PDF	HALDB
Index list data set	NO	YES
Extended pointer	NO	YES
Self-healing pointers	NO	YES
Dynamic allocation	YES	NO
New IMS status codes	NO	YES

Parallel Processing

Batch and BMP applications that read large databases can improve their performance by reading partitions in parallel. Both PDF and HALDB allow a database PCB to only read a selected partition by adding a JCL statement to the batch or BMP application. Running an application for each partition in parallel will greatly improve sequential read performance. The same can be done for batch or BMP programs that are driven by input files if the input can be separated for each partition to allow the programs to run in parallel.

Pointer Updates

HALDB self-healing pointers are designed to allow HALDB database or partition reorganization without updating the secondary indexes or logically related databases, which does improve reorganization performance. However, the pointers in indexes and logical relationships must be updated and healed eventually, and that normally happens when the first BMP with update intent reads the database from the secondary index or logical relationship.

The pointer will not be healed until an application with update intent can update the pointer. If your applications read the secondary index only with read-only BMPs, you must run an update BMP to heal the secondary index. Databases with a high-percentage of logical relationships between databases could experience performance problems with BMPs healing the pointers.

The number of locks will dramatically increase when the pointers are healed. The locks are not released until the application reaches a commit point. BMPs with update intent that do not take frequent checkpoints may hold so many locks that online performance can be affected.

Support for	PDF	HALDB
DBRC Required	NO	YES
Partition independence	NO	YES
Image copy indexes	YES	NO
Image copy secondary indexes	YES	YES (PSINDEX)
Index list data set	NO	YES
Online reorganization tool	Database Director	HALDB Online Reorganizer
Clone partitions	Database Director	NO

Database Maintenance and Test Issues

Key areas of comparison between HALDB and PDF include the online reorganization tools available for each product, the HALDB requirement for DBRC, partition independence, and the ability to archive the partitioned database.

Reorganization Utilities

PDF ships with the NEON Eclipse Reorganization Utilities/PSE, which provide an optimal database maintenance suite for PDF databases. The utilities provide a consistent level of performance, regardless of the organizational status of the database. They process multiple partitions in parallel, and allow multiple tasks in a single job step. The Eclipse iSurvey utility also allows you to create a high-keys report for an unpartitioned database that you plan to convert to PDF.

HALDB also ships with its own set of HD utilities that provide a way to maintain and reorganize HALDB databases. Because these utilities are separate from other IMS utility sets, they may require additional time to learn and implement. For example, the IMS Load utility or HP Load utility will not initialize HALDB PSINDEXes. They must be initialized separately with the HALDB Partition Data Set Initialization Utility.

Online Reorganization Tools

HALDB online reorganization (OLR) is new in IMS Version 9.1. The partition is reorganized into a partner partition data set, using segment locking and logging. Each segment insert into the partner data set is logged-approximately three log records are required for each segment. IBM admits that the amount of log data is voluminous. The log volume and buffer contention during OLR can affect online performance. No image copies can be taken during while OLR runs, and stopping the online reorganization can be difficult once it starts.

Because ILDS and self-healing pointers are used for OLR, the performance issues that are incurred with healing pointers after a reorganization also apply to HALDB OLR. That is, you may need to create a BMP with update intent to heal the pointers, and performance can take a significant hit if a number of logical relationships and index pointers need to be healed at any given time. Finally, after OLR completes, it can be difficult to replicate the reorganized database, because of HALDB data set name requirements and a complicated setup for the RECONs.

PDF requires a separate product, Database Director (D^2), for online reorganizations. D^2 provides an enormous performance improvement (both in run time and CPU used) to the online reorganization utility supplied with HALDB. D^2 reorganizations handle all IMS commands required during the process, and incur significantly less overhead than the HALDB OLR. You can also terminate a D^2 reorganization at any time, rolling back the operation with no impact to your system.

Image Copy for Test Data

If you use IMS or DF/DSS image copies to copy production data for test purposes, or if you use shadow databases for processing outside of the production environment, you may need to consider your choice of partition tool carefully. While PDF has no restrictions on this process, the HALDB partition ID and primary index do affect using image copies for this purpose.

The HALDB ID is assigned by DBRC when the partition is defined. It is stored in the first block of the partition data set and in the DBRC RECON. These ID records must match when the partition is opened by IMS. The partition ID of a HALDB image copy is unlikely to have a corresponding ID in the RECON where the test or shadow database is defined. Therefore, you cannot copy image copies of HALDB partitions.

The IBM workaround to this problem is to export the DBRC RECON definition and import it into the destination RECON. Copying databases by using HD unload files instead of image copies to copy a HALDB database also circumvents the problem, because the reload process stores the DBRC partition number from the RECON in the first block of the partition data set.

Test Environments

Because PDF is designed on the model of adding a data set group, it does not impact your IMS test environment in a significant way. PDF databases can be registered with DBRC, but that is not a requirement. The test environment and applications need not change for PDF databases.

HALDB can increase the complexity of your test environment in a number of ways. You may need additional test resources because each tester will require their own set of DBRC RECON data sets, HALDB databases, and a high-level qualifier. IBM also recommends restoring the database or creating a new set of RECON data sets for each test run.

Returning to an Unpartitioned Format

Because to the increased size of the PSINDEX (up to three times the size of an upartitioned secondary index) and the additional data sets and logging required by HALDB, returning to an unpartitioned database can be extremely complex. And by returning the additional data to an unpartitioned state, there may be new demands on a system that is already strained for capacity. Any application changes would also have to be backed out, requiring additional time and effort to complete the return.

By contrast, PDF adds no additional data structures to the system, and backing out the partitions is as simple as reversing the conversion process. A DBDGEN and ACBGEN will be required. Since no application changes are made to convert to PDF, there is no need to consider applications in the return.

In fact, it might be simpler to convert a HALDB database to PDF, and then return the PDF database to an unpartitioned state, than to try and return HALDB directly to an unpartitioned database.

Partition Independence

The most significant HALDB feature that PDF does not have is partition independence. PDF DBRC support is the same as data set groups; consequently, DBRC authorization is done for the entire database. HALDB DBRC authorization is done for each partition. HALDB partitions can be /DBR'd to be used by database utilities while the other partitions are available online.

However, you might need to change your applications to take advantage of partition independence. DEDBs have always had area independence, and applications were designed to take advantage of the FH status code (area not available). Full-function applications were not designed for an equivalent database record not available status code. IMS Version 2 added the NA status code in the DBPCB to indicate a database was not available; however, the NA status code is not set for unavailable HALDB partitions.

An application does not know that a HALDB partition is unavailable until it makes a call to the partition and receives a BA status code. Most applications do not issue the INIT command for extended status codes, so the call to an unavailable partition will result in a U3303 pseudo-abend. Your customer may wonder what happened to the request since all transactions against the partition are suspended until the partition is restarted.

Archiving the Database

If you use database reorganization to archive and purge inactive segments, it might be important to you that HALDB does not update secondary indexes during reload. Source segments cannot be deleted during reorganization because HALDB secondary indexes contain index records for source segments that have been deleted during reorganization, and deleting the source segments would result in an index pointer error. This design precludes the kind of common practice in which segments are deleted from the database during reorganization by a user exit or by modifying the unload data set.

Another archive approach is to keep the most current data in the first (or last) partition and then age the data by incrementing partition numbers during reorganization, with the oldest partition data being discarded or archived. PDF provides complete flexibility for this technique. HALDB is less cooperative.

Conclusion

There are times when partitioning provides the optimal solution for managing very large databases, and both IBM and NESI provide partitioning solutions to accomplish the job. Before choosing a solution, however, you should carefully consider your current IMS structure and requirements, the cost of conversion, the ease of implementing both test and production environments with partitioned databases, and the ongoing maintenance requirements for any partitioned environment. In many cases, PDF provides an easier implementation with superior performance and a more flexible choice for both test and maintenance requirements.

Additional Reading

For more information about capacity management issues, we recommend the following publications:

Large Database Alternatives, by William Keene, available at http://www.neonesoft.com

Managing Large IMS Databases with Partitioning Products, by Don Coltrane, available at http://www.neonesoft.com

The Complete IMS HALDB Guide: All You Need to Know to Manage HALDBs, Jouko Jäntti et al., available at http://www.ibm.com

About the Author

Tom McMillan is the product author of Partitioned Database Facility (PDF). He has more than 20 years of IMS experience, including 12 years at IBM as a systems engineer and IMS database developer and 8 years designing and developing products for NEON Enterprise Software. At IBM's Santa Teresa Lab, he led the design and development of IMS 3.1 Change Data Capture for IMS to DB2 data propagation. He holds a B.A. degree in business economics from the University of California at Santa Barbara.

white paper

About NEON Enterprise Software

NEON Enterprise Software is the technology leader in enterprise data availability software and services. In a world where every second counts, our tools maximize database performance and availability and minimize business risk. Founded in 1995, NEON Enterprise Software is headquartered in Sugar Land, Texas and serves customers worldwide with its dedicated team of industry experts.

For more information, visit www.neonesoft.com or call 281.491.6366 or 888.338.6366.

Copyright ©2007 NEON Enterprise Software, Inc. All rights reserved. Eclipse iChange, Eclipse iCheck, and Eclipse iRecover are registered trademarks of NEON Enterprise Software. Database Director, EADO, Eclipse iBuild, Eclipse iCopy, Eclipse iExtend, Eclipse iExtract, Eclipse iLM, Eclipse iLoad, Eclipse iRepair, Eclipse iSurvey, Eclipse iUnload, Eclipse Reorganization Utilities, HALO, iServe, iServe DBA, iServe SP, Lightning DEDB, Lightning Extend Instant, Lightning Extend Online, Lightning Utilities, Lightning X, Mission Control, NESS, Record Reorganizer and TITAN Archive are trademarks of NEON Enterprise Software. PDF is a trademark of NEON Systems, Inc., in the USA and in other select countries, and is licensed to NEON Enterprise Software. All other trademarks are the property of their respective owners. 8/07

white paper