An Introduction to Oracle Partitioning

An Oracle White Paper
December 2005

ORACLE

INTRODUCTION

The basic concepts of partitioning pervade our everyday life. What would a database be like without partitioning? Imagine going to a bookstore, where the books weren't organized onto shelves by themes. Instead, when new books arrived at the bookstore, they are placed into any open slots on any shelf. Even if the bookstore had an online index, which could tell you exactly where each individual book was located, this would still be a terribly inconvenient system. If you wanted to find ten books on the subject of "Oracle Partitioning", you would need to go to ten completely different areas of the bookstore. A database without partitioning is like a bookstore without a shelving structure. Partitioning helps to organize the data within a database, so that similar data is grouped together to simplify data access and data management.

Oracle Partitioning is an option to Oracle Database 10*g* Enterprise Edition. At a technical level, Oracle Partitioning allows tables and indexes within a database to be subdivided into smaller pieces. Oracle can 'partition' a table by ranges of values, so for example the data for Jan 1, 2006 through Jan 31, 2006 is in its own partition. Oracle can also partition a table by a list of values, so that North American data is in one partition, while the European data is in another partition. And Oracle can partition along multiple dimensions, such as partitioning by date along one dimension, and by region along another dimension. Oracle provides several other partitioning schemes, to accommodate all types of business scenarios.

Oracle Partitioning is embedded tightly into the core database engine and supported by Oracle's administrative tools. From an application perspective, partitioning is completely transparent: no changes need to be made to the application or to the SQL statements in order to use partitioning.

This paper will describe the typical usages and benefits of partitioning, illustrating how partitioning can benefit any type of database application, from OLTP to DSS to content management, at any scale, from megabytes to petabytes.

VLDB AND DATA WAREHOUSING

Partitioning is a critical feature for managing large databases. Growth is the basic challenge that partitioning addresses for large databases, and Oracle Partitioning enables a "divide and conquer" technique for managing the large tables in the database, especially as those tables grow.

Although your database may have twice as much data next year as it does today, your end-users are not going to tolerate their application running twice as slow, your database is not going to be given twice as much time to complete maintenance and batch processing, and your IT managers are not going to double your hardware budget. Partitioning is the feature that allows a database to scale for very large datasets while maintaining consistent performance, without unduly increasing administrative or hardware resources.

How does this work? Returning to the bookstore analogy, suppose that you walk into a bookstore, looking for books on Oracle databases. You simply walk to the aisle containing computer books or database books. While a small bookstore may only have 5 aisles, and a large bookstore may have 100 aisles, you will always be able to find the section containing the books about Oracle databases in a few seconds. Even though the large bookstore may be 20 times larger, your search for database books takes roughly the same amount of time as in a small bookstore.

Just as the grouping of books by subject makes it easy and fast to find books in even the largest bookstores, partitioning enables faster data access within an Oracle database. Whether your database has 10GB or 10TB of data, Oracle Partitioning can speed up data access by orders of magnitude.

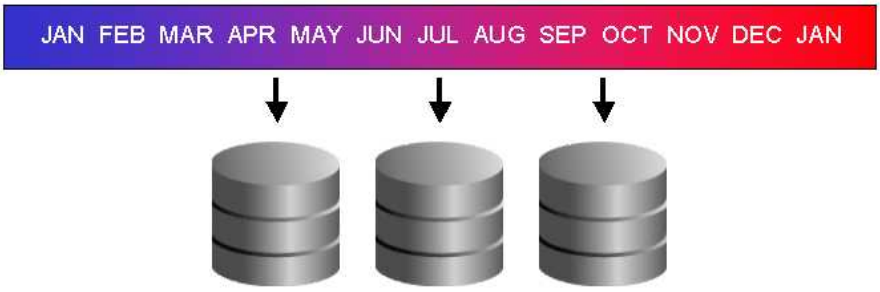
PARTITIONING AS THE FOUNDATION FOR INFORMATION LIFECYCLE MANAGEMENT

Information Lifecycle Management (ILM) is a set of processes and policies for managing data throughout its useful life. Practically, one important component of an ILM strategy is determining the most appropriate and cost-effective medium for storing data at any point during its life time: newer data used in day-to-day operations is stored on the fastest, most highly-available storage tier, while older data which is accessed infrequently may be stored on a less-expensive and less-performant storage tier.

Oracle Database 10*g* provides the ideal environment for implementing your ILM solution. Oracle supports multiple storage tiers, and since all of the data remains in the Oracle database, the use of multiple storage tiers is completely transparent to the application and the data continues to be completely secure. For more details on ILM, please see
http://www.oracle.com/technology/deploy/ilm/index.html.

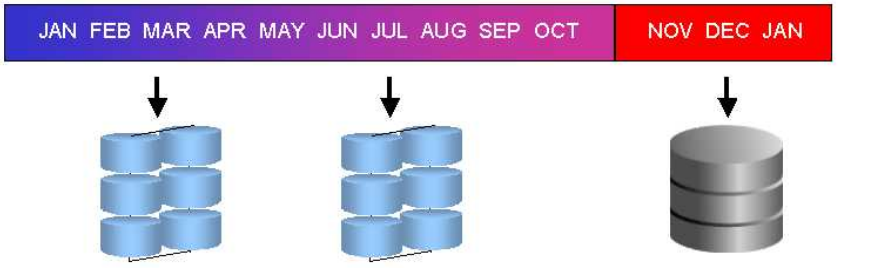
One of the benefits of implementing an ILM solution is to reduce costs, by leveraging appropriate storage tiers, while maintaining all of the data required for

business or regulatory purposes. According to a recent report[1], the cost per gigabyte for high-end storage solution ranges from $18.90 to $34.90, while the cost per gigabyte for low-end storage ranges from $5.90 to $10.40. For a database with several terabytes of data, the cost savings can be considerable. Let's suppose that we had a large database containing 15 months of sales data, which requires 2.5 TB of storage. Since much of our recent sales data is mission critical, we might decide to store all of the sales data on high-end storage. If we are paying $25/GB for this storage, then our storage cost is $25 * 2.5 * 1000 = $62,500.



All Data on High-End Storage

Compare this to an ILM approach with multiple storage tiers. Based upon our ILM policies, we have designated the most recent 3 months (500GB) of this data as the most critical business data, while the previous 12 months (2TB) of data is used infrequently. If our low-end storage cost is $7/GB, then our total storage cost for this configuration is ($25 * .5 * 1000) + ($7 * 2 * 1000) = $12500 + $14000 = $26,500. The underlying storage for the ILM-based solution is almost 2 ½ times less expensive -- and this cost saving is realized without compromising any of the key service levels of the application since the critical data is still stored on high-end storage.



Older Data on Low-End Storage

New Data on High-End Storage

Partitioning is the capability that enables this ILM scheme to be implemented within the database. If your sales data was partitioned by month, then you would store the most recent three months of data (corresponding to three partitions) on the high-end storage, and the older 12 months of data (12 partitions) on low-end

[1] "Disk Storage Price Forecast, 2H04: Prices will Fall Through 2005, Jan 2005, Stanley Zaffos, Gartner

storage. The database administrator could implement this ILM scheme without any modifications to the application itself, since the sales data is accessed in exactly the same way, whether it is entirely on high-end storage or spread across multiple storage tiers.

Further savings can be realized by using partitioning in conjunction with other Oracle database capabilities. For example, in addition to storing older data on less expensive storage, the older data could also be compressed thus further reducing storage costs. The newer data (in separate partitions) remains uncompressed to optimize access and updates.

PARTITIONING FOR EVERY DATABASE

The benefits of partitioning are not simply for very large databases; every database, even small databases, can benefit from partitioning.

While partitioning is a necessity for the largest databases in the world, partitioning is obviously beneficial for the smallest database as well. If you have just a few shelves of books in your home library, you will still find it useful to organize your books by subject matter. In that manner, even a database whose size is measured in megabytes will see the same type of performance and manageability benefits from partitioning as the largest multi-terabyte systems.

And, although multiple storage tiers and sophisticated ILM policies are most often found in enterprise-level systems, all companies and all databases need some degree of information lifecycle management. The most basic of ILM operations, archiving older data and purging or removing that data from the database, can be orders of magnitude faster when using partitioning.

CONCLUSION

Oracle Partitioning makes it easier and faster to manage any amount of data. Just as a bookstore organizes its books on shelves by subject area, Oracle Partitioning allows you to organize your database to simplify administration and improve performance. Whether you are building a large data warehouse or supporting a modest operational database, Oracle Partitioning can help your application meet its performance requirements with less administrative resources and less hardware resources.

An Introduction to Oracle Partitioning
December 2005
Author: George Lumpkin
Contributing Authors: Lilian Hobbs, Hermann Baer

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com