Oracle9*i* Database Architecture on Windows

An Oracle Technical White Paper
August 2001

ORACLE

Oracle9*i* Database Architecture on Windows

Oracle9*i* Database Architecture on Windows

EXECUTIVE OVERVIEW

With the introduction of the Oracle9*i* Database for Windows, Oracle once again provides the enterprise scalability, reliability, and high performance that customers require. The Oracle9*i* Database provides enterprise-class data solutions through tight integration with the advanced features of the Windows operating system and hardware. By using a native thread-based Windows service model, Oracle9*i* ensures high performance and scalability. Oracle can meet enterprise hardware requirements through large and raw file support, large memory support, and parallel computing via clustering. Future performance and scalability enhancements will be available with the impending release of the 64-bit version of Oracle9*i* on Windows. This paper discusses how the Oracle database architecture has been designed to take full advantage of advanced Windows operating system features and hardware.

INTRODUCTION

As the adoption of Windows takes place, Oracle9*i* has become the market leading database for the Windows platform. From the outset, Oracle's goal has been to provide the highest performing and most tightly integrated database on Windows. As a result, Oracle invested early to move its market leading UNIX database technology to the Windows platform. In 1993, Oracle was the first company to provide a database for Windows NT.

Initially, Oracle's development efforts were concentrated on improving the performance and optimizing the architecture of the RDBMS on Windows. Oracle7 on Windows NT was redesigned to take advantage of several features unique to the Windows platform, including native thread support and integration with some of the Windows NT administrative tools, such as the Performance Monitor and the Event Viewer.

However, Oracle9*i* on Windows has evolved from basic operating system level integration to more advanced service integration in the Windows platform, including the upcoming 64-bit version of Windows. As always, Oracle continues to innovate and to leverage new Windows technologies. This white paper discusses the Oracle9*i* database architecture on Windows in detail.

ORACLE9*i* ARCHITECTURE ON WINDOWS

When running on Windows, Oracle9*i* contains the same features and functionality as it does on the various UNIX versions of Oracle. However, the interface between Oracle9*i* and the operating system has been substantially modified to take advantage of the unique services provided by Windows. As a result, Oracle9*i* on Windows is not a straightforward port of the UNIX code base. Significant engineering work has been done to make sure that Oracle9*i* fully exploits Windows while guaranteeing that Oracle9*i* is a stable, reliable, and high performing system for building applications.

Thread Model

Compared to Oracle9*i* on UNIX, the most significant architectural change in Oracle9*i* on Windows is the conversion from a process-based server to a thread-based server. On UNIX, Oracle uses processes to implement background tasks, such as database writer (DBW0), log writer (LGWR), Multi-Threaded Server (MTS) dispatchers, and MTS shared servers. In addition, each dedicated connection made to the database causes another operating system process to be spawned on behalf of that session. On Windows, however, all of these processes are implemented as threads inside a single, large process. What this means is that for each Oracle database instance, there is only one process running on Windows for the Oracle9*i* server itself. Inside that Oracle process are many running threads with each thread corresponding directly to a process in the UNIX architecture. So, if there were 100 Oracle processes running on UNIX for a particular instance, that same workload would be handled by 100 threads in one process on Windows.

Operationally, client applications connecting to the database are unaffected by this change in database architecture. Every effort has been made to ensure that the database operates in the same way on Windows as it does on other platforms, even though the internal process architecture has been converted to a thread-based approach.

The original motivation to move to a thread-based architecture on the first version of Windows NT was to improve database performance when dealing with files shared among processes. By simply converting to a thread-based architecture and modifying no other code, performance was dramatically increased by avoiding this Windows operating system bottleneck. Although the original motivation for the change is no longer present, the thread-based architecture for Oracle remains because it has been proven to be very stable and maintainable. In addition, there are other benefits that arise out of thread-based architecture. These benefits include faster operating system context switches among threads (as opposed to processes); a much simpler Shared Global Area (SGA) allocation routine which does not require the use of shared memory; faster spawning of new connections since threads are more quickly created than processes; and decreased memory usage since threads share more data structures than processes do.

Internally, the code to implement the thread model is compact and very isolated from the main body of Oracle code. Robustness has been added to the architecture through the use of exception handlers and through routines used to track and de-allocate resources. Both of these additions help allow for 24x7 operation with no downtime due to resource leaks or an ill-behaved program.

Services

In addition to being thread-based, the Oracle9*i* database is also not a typical Windows process. It is a Windows *service*, which is basically a background process that is registered with the operating system. The service is started by Windows at boot time and runs under a particular security context. Oracle's conversion into a service was necessary to allow the database to come up automatically upon a system reboot, since services require no user interaction to start.

When the Oracle database service starts, there are none of the typical Oracle threads running in the process. Instead, the process waits for an initial connection and startup request from SQL*Plus which will cause a foreground thread to start and which will eventually cause the creation of the background threads and the SGA. When the database is shutdown, all the created threads will terminate, but the process itself will continue to run and wait for the next connection request and startup command.

In addition to the Oracle database service, administrators can automatically spawn SQL*Plus to start up and open the database for use by clients. Finally, the Oracle Net Listener is a service since it too needs to run before users can connect to the database. Although these implementation details do not affect how clients connect to or otherwise use the database, this information is very relevant for Windows database administrators.

File I/O Enhancements

Oracle9*i* on Windows includes support for large files and for raw files. In an effort to guarantee that all features of Windows are fully exploited by Oracle9*i*, the database supports 64-bit file I/O to allow the use of files larger than 4GB in size. In addition, physical and logical raw files are supported as data, log, and control files in order to enable Oracle Real Application Clusters on Windows and to maximize performance.

64-bit File I/O

Internally, all Oracle9*i* file I/O routines support 64-bit file offsets, meaning that there are no 2GB or 4GB file size limitations for data, log, or control files as is the case on some other platforms. In fact, the limitations that are in place are generic Oracle limitations across all ports. These limits include 4 million database blocks per file, 16KB maximum block size, and 64K files per database. If these values are multiplied, the maximum file size for a database file on Windows is calculated to be

64GB while the maximum total database size supported (with 16KB database blocks) is 4 petabytes, or 4000 terabytes.

Raw File Support

Like UNIX, Windows supports the concept of raw files, which are unformatted disk partitions that can be used as one large file. Because they are unformatted partitions, raw files have the benefit of no file system overhead. As a result, a slight performance gain can occur by using raw files for database or log files. However, the downside of using raw files is fewer manageability options since standard Windows commands do not support manipulating or backing up raw files. As such, raw files are generally used only by very high-end installations and by Oracle Real Application Clusters, where they are required.

To use a raw file, Oracle requires the filename specifying the drive letter or partition to use for the file. For instance, the filename "\\.\PhysicalDrive3" tells Oracle to use the 3rd physical drive as a physical raw file as part of the database. Likewise, "\\.\G:" tells Oracle to use the logical raw file that has been assigned drive letter G. Also, a file such as "\\.\log_file_1" is an example of a raw file that has been assigned an alias to make user recognition of the file easier. Aliases can be assigned with the SETLINKS utility provided with Oracle9*i*. When specifying raw filenames to Oracle, care must be taken when choosing the partition number or drive letter, as Oracle will simply overwrite anything on the specified drive when it adds the file to the database. Overwriting will occur even if it is already an NTFS or FAT formatted drive.

To Oracle, raw files are no different from other Oracle database files. They are treated in the same way by Oracle and can be backed up and restored via Recovery Manager as any other file can be.

Scalability Enhancements

One of the key goals of the Oracle9*i* database on Windows is to fully exploit any technologies that can help increase scalability, throughput, and database capacity. The following section describes a few of these technologies, how they affect Oracle, and the benefits that can be derived from them.

4GB RAM Tuning (4GT) Support

Windows 2000 Advanced Server and Datacenter Server include a feature called 4GB RAM Tuning (4GT). This feature allows memory-intensive applications running on Windows 2000 to access up to 3GB of memory as opposed to the standard 2GB that is allowed in other editions of Windows 2000. The obvious benefit to Oracle9*i* is that 50% more memory becomes available for database use. More memory can increase SGA sizes or connection counts. All Oracle database server releases since Oracle7 version 7.3.4 have supported this feature. No modifications are necessary to a standard Oracle installation to use this feature. The

only configuration change required is to ensure that the /3GB flag is used in the Windows 2000's boot.ini file.

Large User Populations

Oracle has made great efforts to support large numbers of connected database users on Windows 2000. As far back as Oracle7 version 7.2, customers have deployed applications with over 1000 concurrent connections to a single database instance on Windows NT in a production environment. As time has progressed, that number has increased to a point where well over 2000 users can be concurrently connected to the database in production environments. When using the Oracle Multi-Threaded Server architecture, which limits the number of threads running in the Oracle database process, over 10,000 simultaneous connections to a single database instance have been accomplished. Additionally, network multiplexing and connection pooling features can allow a large configuration to achieve more connected users to a single database instance. Finally, Oracle Real Application Clusters can be used to again increase connection counts dramatically by allowing multiple server machines access to the same database files. Clusters increase capacity for user connections and, at the same time, increase throughput.

64-bit Support

The next leap in Oracle9*i* performance and scalability on Windows will occur with the introduction of a 64-bit version of Oracle9*i*. 64-bit Oracle9*i* will run on Intel Itanium processor-based machines and the 64-bit version of Windows. Oracle development teams have been working closely with these technology vendors to guarantee that Oracle9*i* on Windows will be released in production form soon after the hardware and operating system become generally available. As with other 64-bit Oracle ports to different UNIX variants, a 64-bit port of Oracle9*i* to Windows will be able to handle more connections, allocate much more memory, and provide much better throughput than the current version of the database on Windows. The migration path from 32-bit to 64-bit Oracle will be very straightforward. There will be no need to recreate databases, nor will a full export and import be required. All that will be required will be to copy the current data files to the new system, install 64-bit Oracle9*i*, start the database as normal, and run a few SQL scripts to update the data dictionary.

From an architectural perspective, the current proven thread-based architecture will be used for the 64-bit port. For Oracle's development teams, very little new code will need to be written during the move to 64-bit since the underlying operating system APIs are expected to remain substantially the same. Since Oracle9*i* has already been ported to other 64-bit operating systems, Oracle's experience with 64-bit technology will ensure customers have a high quality, stable 64-bit database on Windows.

CONCLUSION

In summary, Oracle's database on Windows has evolved from a port of its UNIX database server to a well-integrated native application that takes full advantage of Windows services, features, and underlying hardware. Oracle continues to improve the performance, scalability, and capability of its database server on Windows, while at the same time producing a stable, highly functional platform on which to build applications. Oracle is fully committed to providing the highest performing, best-integrated database on the Windows platform for both 32-bit and 64-bit versions.

For further information on Oracle's Windows products, please visit

http://www.oracle.com/nt

http://otn.oracle.com/tech/nt

Oracle9*i* Database Architecture on Windows
August 2001
Author: David Colello
Contributing Authors: Alex Keh

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
www.oracle.com

Oracle Corporation provides the software that powers the internet.