Administer the Oracle7 Server for Windows NT

Brian Guza
ISCG Inc.

Introduction

Microsoft Windows NT is gaining momentum as a strategic operating system for client-server systems, and Oracle7 Server for NT is becoming the database of choice on this platform. This paper provides information related to the installation, configuration, and administration of an Oracle7 database on the Windows NT platform. Specifically, the following topics will be discussed:

- Working with the 'starter' database that is created by the Oracle7 Server for NT installation procedure.
- Creating a production-ready database "from scratch" using Oracle7 Server for NT.
- Database Startup.
- Database Backup.
- A case for using OFA to administer NT databases.
- OFA guidelines for the NT operating system.

Installation and Starter Database

The Oracle7 Server for NT RDBMS installation process is extremely simple and reliable. The Oracle NT installation methodology is nearly identical to the GUI-driven, user-friendly process of the Microsoft Windows Installer. By choosing "Oracle7 Server" from the Oracle Software Asset Manager and answering a few basic questions, an entire server installation can be performed on a Pentium-class machine in about twenty minutes.

The Oracle7 Server installation procedure creates a "starter" database as part of the installation process. At the end of the server installation, this database is completely functional and can even accept SQL*Net connections without additional user setup. This database requires little or no database administration and is designed for users working on small, non-critical applications. Although this database is fully functional, it is not capable of supporting an OLTP production database.

Several of its deficiencies include:

- Two control files are created, but both files reside on the same disk device.
- Only two redo log files are created, and no redo log mirroring is performed.
- All data files are created on the same disk device.
- The database is created in NOARCHIVELOG mode.
- Backup utilities for Oracle7 Server for NT exist, but none are activated after the server installation.
- Rollback and Temporary segment space is not adequate for larger databases.
- SGA and memory parameters are not adequate for larger databases.

Based on this configuration, the starter database does not appear to be production-ready "out of the box." Steps must be taken to reconfigure the starter database or to create new databases that are more suited to store and manage production data. In most cases, it is easier to create a new database from scratch rather than it is to reconfigure the starter database.

Database Creation

A fully-configured Oracle7 NT environment is very similar to an Oracle7 environment on other operating systems in terms of basic configuration and best practices for database creation and administration. Most of the database objects in an Oracle7 database (control files, redo logs, SGA, tablespaces, etc.) are present in an NT database environment. The standard remedies for resolutions of "normal" database creation issues apply to Oracle NT (i.e., set the CONTROL_FILES parameter in INIT.ORA while database is down to move control files).

The steps required for the creation of an Oracle7 database on NT are somewhat similar to the steps that would be required on most operating systems. The following brief list of tasks should be performed to create

an Oracle7 Server for NT database:

1. Create a new NT service called OracleServicexxxx where xxxx is a four or less character SID identifier. This is accomplished by invoking the Oracle Instance Manager in NT command mode. If the service is created with the full-screen implementation of the Instance Manager, this utility creates database files (log files, system tablespace, etc.) in addition to the Oracle service. The syntax for creating a new service from an NT command line would be similar to the following:

```
C:\>  ORADIM73  -NEW -SID db1  -INTPWD
psw_for_internal  -MAXUSERS some_number
```

2. From an NT Command line, set the ORACLE_SID environment variable to the SID name using the NT SET command.

```
C:\>  SET ORACLE_SID=xxxx
```

3. Copy the INIT.ORA template file to INITsid.ORA in the database directory of ORANT (the ORANT\DATABASE directory for NT 7.2 and 7.3 and the ORANT\RDBMS71 directory for 7.1).

```
C:\orant\database> COPY init.ora initxxxx.ora
```

4. Copy or create a script to issue an Oracle CREATE DATABASE command. Use NT directory structure for the datafiles in the script. Use mirrored redo-log files in this command if needed. Execute the CREATE DATABASE command from SVRMGR or SQLDBA in line mode.

5. Run the CATALOG.SQL, and optionally the CATPROC.SQL (for PL/SQL "option") and CATREP.SQL (for Advanced Replication Option) from the \ORANT\RDBMSxx\ADMIN directories.

6. Write and execute SQL scripts to create tablespaces and rollback segments for the new database. All "best practices" that apply to separating indexes from tables, temporary data from everything else, etc. are applicable to NT.

7. Set ARCHIVE parameters if necessary. Test the archive creation process with ALTER SYSTEM SWITCH LOGFILE to ensure that these INIT.ORA parameters are set correctly.

8. Create known user accounts in the Oracle database. Note: As with Unix and VMS, it is not necessary to create NT user accounts in order for users to access data in an Oracle NT database.

9. Add entries for the database in LISTENER.ORA and TNSNAMES.ORA in \ORANT\NET-WORK\ADMIN and \ORAWIN\NETWORK\ADMIN with Network Manager, Easy SQL configuration, or manually, if necessary.

Startup

It is normally preferable to start all databases on an NT machine automatically after a server has been rebooted. Although this process is relatively simple on the UNIX and VMS implementations of Oracle, this has been a problem area for Oracle NT that is still present in Version 7.3.

In versions prior to 7.2, a user was expected to start the database manually with the Instance Manager each time that the NT server was rebooted. In the 7.2 and 7.3 versions of Oracle NT, hooks were added to create an OracleStartxxxx service that starts the database automatically when the NT server is rebooted. Although this is a handy feature,, this method requires that the INTERNAL password be written to a text file in order for the service to start the database.. For a secure database with sensitive information, even the most current approach does not appear to be sufficient.

To create a safe, reliable database startup script, it is necessary to write a script that utilizes the SYSOPER capabilities of Oracle. The SYSOPER role allows the grantee to start and stop the database without having privileges to read the actual data in the database. Although the password associated with an Oracle account with SYSOPER must be written to a file, this account will not have access to the data in the database. If the INTERNAL password is found on disk by an unauthorized user, all data and data structures in the database can be read, changed, or deleted from this account.

By adding a more secure script to NT startup, databases can be started automatically without any breaches in security. To implement this approach,, complete the following steps:

1. Change the REMOTE_LOGIN_PASSWORDFILE parameter to EXCLUSIVE in the INIT.ORA file of all databases on an NT server. Restart the databases if necessary. This step is necessary to utilize the SYSOPER role of Oracle7.

2. Create a user called SYSOP with a non-obvious password. Grant SYSOPER to the SYSOP account.

3. Create an NT command file and SQL script similar to the one in the figure 1.

DBSTART.BAT

```
rem Start the Listener service only if it has not been started already.

net start | find /I "OracleTNSListener" || net start OracleTNSListener

rem Start the GEN1 Database with Server Manager (use SQLDBA for pre-7.3 Databases)

set ORACLE_SID=GEN1
net start | find /I "OracleServiceGEN1" || net start OracleServiceGEN1
c:\orant\bin\svrmgr23 command=@c:\oradba\local\dbstart.sql

rem Start the SID2 Database

set ORACLE_SID=SID2
net start | find /I "OracleServiceSID2" || net start OracleServiceSID2
c:\orant\bin\svrmgr23 command=@c:\oradba\local\dbstart.sql
```

DBSTART.SQL

```
spool c:\oradba\local\strt%ORACLE_SID%.lis
connect sysop/operator123 as sysoper;
startup
exit
```

Figure 1.

4. Use REGEDT32.EXE to add the startup file to the NT server startup by appending the full filespec of the command file to the following registry entry: HKEY_LOCAL_MACHINE\Software\Microsoft\WindowsNT\CurrentVersion\WinLogon\System

Backup

The issues related to the backup and recovery of an NT database are similar to the issues associated with most other operating systems. Oracle datafiles belonging to open databases should not be backed up at the operating system level because they are not reliable for recovery. Additional processing must be added to an NT backup script to ensure that Oracle recovery can be performed successfully.

The traditional methods of backup can be adapted to the NT operating system quite easily. These methods include:

- Cold Backup: All databases are stopped before operating system backup and restarted after the backup has completed.
- Hot Backup: The ALTER TABLESPACE BEGIN/END BACKUP commands are used to manage backup while keeping the databases available during the actual backup process. Databases must be in ARCHIVELOG mode in order to implement this method of backup.

After choosing one of these methods, the Oracle commands necessary to implement a reliable backup can be incorporated into an NT command script (similar to a Windows BATCH file). This script can use the NTBACKUP command which is used for disk-to-tape processing When the script has been written and tested, it can be submitted to run automatically at regular intervals (nightly or weekly) by using the AT command of NT (similar to UNIX CRON or VMS SUBMIT). The following figure illustrates a very simple cold backup example using an NT command script and the AT command.

BACKUP.CMD

```
REM  Shut Down Database from server manager (use SQLDBA for pre-7.3 databases)

set ORACLE_SID=GEN1
c:\orant\bin\svrmgr23 command=@c:\oradba\local\dbimmed.sql

REM   Use the NTBACKUP command on c,d,e drives.
REM   Parameters:  Description (/d), Backup with Hardware Compression (/hc),
REM   Type Normal (/t not incremental ),
REM   with Logfile BACKUP.LOG (/l), Include Exeptions only (/e), To Tape 0 (/tape)

NTBackup Backup c: d: e: /d "NT backup of C,D,E " /hc:on /t Normal /l backup.log /e /tape:0

REM   Restart Database. (use SQLDBA for pre-7.3 databases)
REM   Start the service for the database only if is not started already.

set ORACLE_SID=GEN1
net start | find /I "OracleServiceGEN1" || net start OracleServiceGEN1
c:\orant\bin\svrmgr23 command=c:\oradba\local\dbstart.sql
```

SCHEDULE.CMD

```
REM Run BACKUP.CMD each weeknight at 10PM, log and errors to BACKUP.LOG

at 22:00 /interactive /every:M,T,W,Th,F "C:\BACKUP\BACKUP.CMD 1> C:\BACKUP\BACKUP.LOG 2>&1"
```

Figure 2

The Oracle NT Server also includes a backup option that is not available on most operating systems. The OCOPYxx executable in the ORANT\BIN directory can be used to copy an active Oracle datafile to tape or to another disk location. However, there is little or no documentation on this utility, so it is unclear how safe or reliable it would be in a production environment.

OFA on NT?

The Optimal Flexible Architecture (OFA) was developed several years ago by Cary Millsap and the Oracle Performance Group. The goal of the OFA was to provide a standard methodology for the administration and management of Oracle databases that was efficient and flexible.

OFA has become the defacto standard for the installation and administration of UNIX databases. Although it is advertised as platform-independent, the OFA appears to be quite "Unix-centric" with frequent references to "mount points", "/etc directories", and other UNIX objects. There are also several features of OFA that add value exclusively to large installations with many databases. On the surface, it does not appear that it would be advantageous to apply OFA to NT installations.

However, a closer look at OFA reveals several features and benefits that can be applied to the NT environment. Specifically, the following OFA requirements apply directly to NT database administration (from the Oracle7 Administrator's Reference to UNIX):

- The file system must be organized to allow easy administration of growth.
- Hardware costs should be minimized where doing so does not conflict unacceptably with operational considerations.
- Categories of files must be separated into independent directory subtrees, so that files in one category are minimally affected by operations upon files in other categories.
- Administrative information about one database must be separated from that of others.
- Tablespace contents must be separated to minimize tablespace free space fragmentation, minimize I/O request contention, and maximize administrative flexibility.

Based on the these requirements, an NT installation could certainly benefit from a structured methodology that adheres to the principles of OFA. To achieve the goal of adapting NT to OFA, however, some additions and modifications must be made to the existing OFA to account for the different structures and operations of the NT operating system. These modifications would specifically deal with NT issues associated with Account Management, Installation, Remote Management, and File System Requirements. Guidelines that apply to tablespaces, object fragmentation, and other non-operating-system-specific issues would apply directly to Oracle NT and would not have to be rewritten.

A modified OFA methodology is contained in the appendix of this paper. Due to space limitations, only NT-specific additions and modifications have been included in the appendix. As with the existing OFA standard, the NT OFA rules are guidelines only and should only be used when they are applicable to the installation that is being administered.

Conclusion

The Oracle7 Server for Windows NT is a fully-functional implementation of Oracle7. This product is able to take advantage of the power of the Oracle7 RDBMS and the scalability and strategic value of Windows NT. For existing Oracle customers, the Oracle7 Server for Windows NT can provide an efficient, flexible, and cost-effective database environment without incurring the high cost of new product education. For organiza-

tions that do not have the resources (or the patience) to rely on large 'mini-mainframe" machines running UNIX or VMS, Oracle7 for Windows NT is a viable and worthwhile choice for database operations.

Appendix: Oracle NT Optimal Flexible Architecture (OFA) rules

1. NT Administration Standards

1.1 Create an NT user account called ORACLE to install and administer all databases. Grant the NT ADMINISTRATOR privilege to this operating system account.

1.2 Create a LOCAL NT group called ORAadmin. Add the ORACLE account and the personal accounts of all NT users that will be administering Oracle NT databases. Use the ORAadmin group to assign NT file permissions for all Oracle-related files.

1.3 Create a top level directory called ORADATA on each drive that will contain Oracle database files.

1.4 Create a top-level directory called ORADBA on a drive with sufficient disk space to hold all RDBMS administrative scripts for the entire NT Server machine.

1.5 Store Oracle-related files for "secure" databases on NTFS volumes rather than FAT volumes to take advantage of operating system file protections. Use the OraAdmingroup to provide user-level protection on all data and administrative files associated with Oracle, if necessary.

1.6 Separate Database Servers from File Service machines on all NT Servers for optimal performance.

1.7 Configure the Database Server machine with the following parameters: a) Balance Foreground and Background applications. b) Optimize memory for Network Applications rather than File Services.

2. Oracle NT Software Installation Standards

2.1 Use the ORACLE NT user account for database server operations only. Do not use the ORACLE NT account for day-to-day network operations.

2.2 To minimize upgrade efforts, do not create scripts or datafiles in RDBMS version-dependent directory trees.

2.3 Install Oracle NT software on any drive in a top-level directory called ORANT.

2.4 Create an NT file service for the RDBMS software (ORANT) and for Oracle administrative files (ORADBA) for remote administration. Provide FULL ACCESS to these services to the ORAadmin NT group.

3. Oracle NT Database Creation Standards

3.1 Create the following Subdirectories below the ORADBA directory: ADMIN (for database-specific administrative files), LOCAL (for local files that are not database-specific), and PATCH (for software related to patches, upgrades, etc).

3.2 For each database on an NT server, create a directory called ORADBA\ADMIN\sidname where sidname is the name of the database sid to be created.

3.3 Create the following directories under each sid-specific directory: ARCH (archived redo logs), BDUMP (background dump files), EXP (export files). LOGBOOK (informational files), PFILE (parameter file), SCRIPTS (sid-specific SQL files and creation scripts), UDUMP (user dump files).

3.4 Copy the default INIT.ORA file from the correct subdirectory of the ORANT software to the PFILE subdirectory of the correct sid-specific directory. Create a file called INITsid.ORA in the DATABASE subdirectory (RDBMS71 for Oracle 7.1) which contains an IFILE= reference to the sid-specific INIT.ORA file in the ORADBA tree for compatibility with existing Oracle management tools.

3.5 Change the following initialization parameters to point to sid-specific Subdirectories: BACKGROUND_DUMP_DEST (BDUMP), USER_DUMP_DEST (UDUMP), LOG_ARCHIVE_DEST (ARCH).

3.6 Set the REMOTE_LOGIN_PASSWORDFILE parameter to EXCLUSIVE in order to support the use of SYSDBA and SYSOPER privileges.

3.7 Create a subdirectory called ORADATA\sidname on each drive to hold database-related files for an instance.

3.8 Create two or more control files on two separate PHYSICAL drives. Use WINDISK.EXE to determine which virtual drive letters point to which physical drive.

3.9 Restrict all file names to "8 dot 3" notation if WINDOWS systems will be used for remote database management.

3.10 Use existing OFA rules for segment separation, database file names, tablespace names, etc. Current Oracle administrative standards related to these areas on UNIX or VMS hold true for Windows NT.

References

Millsap, Cary, An Optimal Flexible Architecture for a Growing Oracle Database, Oracle Magazine, Winter 1993, pp 41-46.

Oracle, Oracle7 Administrator's Reference for UNIX, Release 7.3, March 1996. Appendix B.

Oracle, Oracle7 Server - Getting Started for Windows NT, Release 7.3, 1996.