

第15章 Windows NT上的联网技术

Windows NT操作系统（这里简称为 NT）在商业环境中正得到广泛的欢迎。NT是一种32位的操作系统，可支持 CISC和RISC体系结构。和UNIX和OpenVMS一样，NT采用全局联网标准，如TCP/IP和SNMP。Oracle公司建议NT用户使用TCP/IP通信协议。TCP/IP上的Net8执行得更好，因为在初始联系以后，不需要发送标准数据，因而可减小数据包的大小。和任何平台一样，NT联网的响应时间由所包含的最慢网络段的速度决定，因此，必须调整网络本身以便Net8能很好地运行。

要了解NT系统上的联网技术，必须先了解 Oracle和Net8在这种环境中是如何工作的。本章将分析在NT系统中进行网络处理的方法。

15.1 Oracle和Windows NT

如果考察一下UNIX系统，就可看到有若干可识别的进程。在下面所示的例子中，-v子句通知UNIX挑选出任何与随后串不相匹配的串。该命令表示查找所有包含 ora但不包含单词LOCAL和grep的进程。在下列所示的该命令的输出中，有9个独立的 Oracle进程和在该UNIX系统上运行的监听程序：

```
ps -ef|grep ora|grep -v LOCAL|grep -v grep

oracle 221 1 0 Jun 15 ? 37:03
/opt/oracle/product/8.1.5/bin/tnslsnr LISTENER -inherit
oracle 258 1 0 Jun 15 ? 307:15 ora_dbw0_cc1
oracle 254 1 0 Jun 15 ? 0:01 ora_pmon_cc1
oracle 260 1 0 Jun 15 ? 0:40 ora_arch_cc1
oracle 262 1 0 Jun 15 ? 26:32 ora_lgwr_cc1
oracle 264 1 0 Jun 15 ? 1:43 ora_ckpt_cc1
oracle 266 1 0 Jun 15 ? 0:10 ora_smon_cc1
oracle 268 1 0 Jun 15 ? 0:02 ora_reco_cc1
oracle 272 1 2 Jun 15 ? 1991:58 ora_d000_cc1
oracle 19971 1 1 Jul 28 ? 197:17 ora_s000_cc1
```

该输出显示了CC1实例的后台进程。NT结构提供了一种内置的多线程性能，因此，一个可执行程序能在同一时间完成许多不同的任务。在 Oracle8i的NT环境中，每个独立的进程在一个称为Oracle.EXE的可执行程序中都作为当前运行的线程来运行（在 Oracle8.0中，这个可执行程序被命名为Oracle80.EXE）。如果使用具有若干线程的同一个可执行程序，那么所有的线程都共享相同的代码、内存空间和其他的结构。在其他的操作系统中，每个独立的进程需要其自己的独立代码实现。由于在一个可执行程序中具有多个线程，Oracle减少了代码和数据段的数量以及处理过程所需要的堆栈，因而减少了所需的内存总量。然而，所执行的任何内存操作对所有的线程都有影响。

图15-1示出了Windows NT Task Manager（任务管理器），带有突出显示的Oracle8i可执行（Oracle.EXE）和Oracle8i监听程序（TNSLSNR.EXE）、Oracle8.0监听程序（TNSLSNR80.EXE）以及Oracle8.0可执行（Oracle80.EXE）。通过同时按下

CTRL_ALT_DELETE键，然后从显示框中选择 Task Manager选项，就可访问 Windows NT Task Manager。

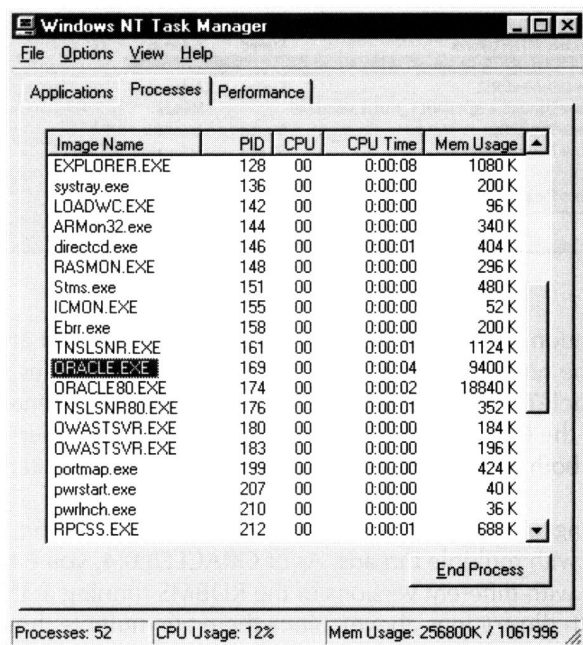


图15-1 Windows NT Task Manager

NT系统中的每个Oracle实例都作为NT能跟踪并在NT启动时启动的独立服务或可执行程序来运行。服务可在后台运行，或者配置成在前台与用户交互。即使没有用户登录到系统，也可启动一个服务并保持运行。这是一种重要的特性，因为数据库服务器不需要依靠一个登录到机器的特定进程来保持数据库或监听程序的运行。

要访问 Services窗口，可从“开始”菜单的“设置”菜单中选择“控制面板”选项，在“控制面板”中选择 Services选项（图标是两个齿轮并标有 Services）。在 Oracle8.0 中，两个独立的服务条目显示在 Services窗口，而在 Oracle8i，则只显示一个条目。图 15-2 示出了 Services窗口，其中突出显示的是 Oracle8i Services ORC1。

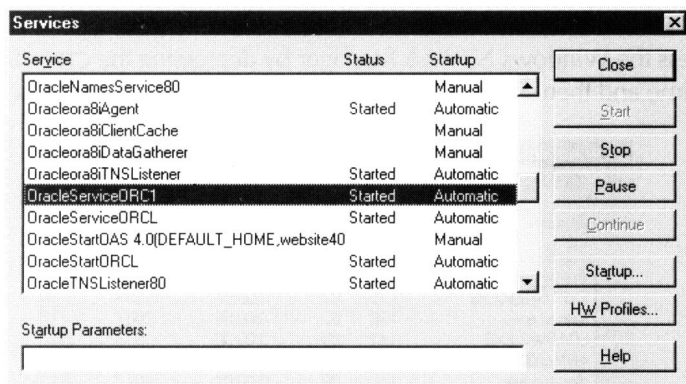


图15-2 Services窗口

在图 15-2 中，可以启动和关闭的 Oracle8.0 服务由 ORCL 来标识。请注意用于 8.0(OracleTNSListener80)和8i(Oracle8iTNSListener)的监听程序的服务条目。8i监听程序服务在其名称中包含有ORACLE_HOME(ora8i)以便于识别。还要注意的，8.0和8i监听程序二者和数据库都已被启动并可供使用。

当操作NT系统上的Oracle时，请记住Oracle使用一个具有多线程的单进程。在 Oracle8.0.4 中，可以有多个具有在同一时间运行的不同版 RDBMS的Oracle宿主（如上述两图所示）。由于每个进程有多个线程，这就意味着不可能轻易识别出哪个线程属于哪个 Oracle进程，因此，必须依靠使用 Services窗口或DOS提示时发出的命令来启动或关闭系统中的监听程序。图 15-3 示出了有某些Oracle线程显示的Performance Monitor（性能监控器）窗口。

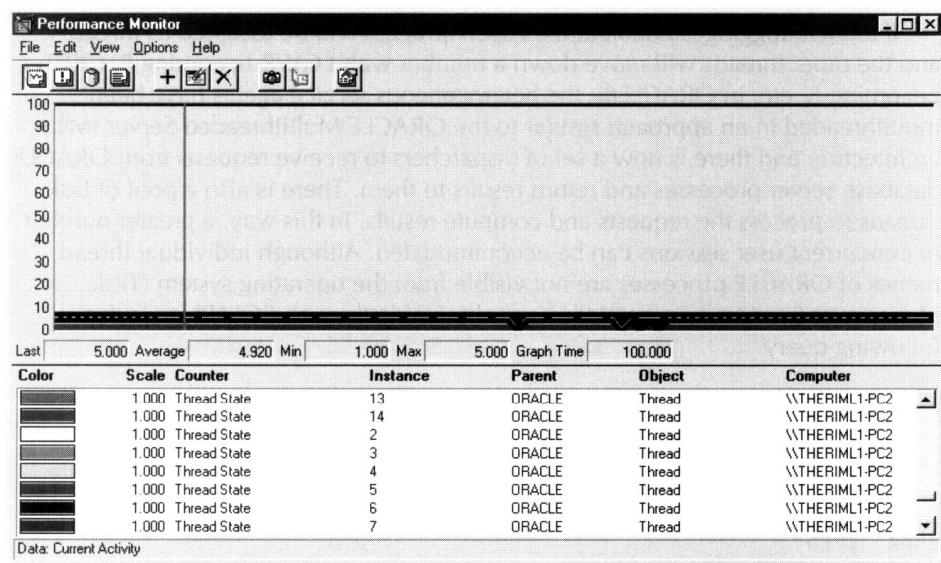


图15-3 Performance Monitor窗口

如图15-3所示，顺序线程号用来标识每个线程，但没提供线程名。Oracle线程以下列顺序来指定：线程 0 是调度程序线程，它处理来自监听程序进程的输入（通过服务启动）；线程 1 是工作者或操作线程，它处理由调度程序线程请求的进程。工作者或操作线程通常以进入专用服务器连接名义启动另一线程。表 15-1 反映了通常在 Oracle8.0和Oracle8i中使用的基本线程号。

表15-1 基本的Oracle线程号

线程名	线程号	线程名	线程号
PMON	2	CKPT	5
DBW0	3	SMON	6
LGWR	4	RECO	7

如果启动归档日志记录，那么 ARCH进程就被赋予线程 4，其他线程下移一个号，即 LGWR变为 5，CKPT变为 6 等等。在 Oracle8i中，以类似于 Oracle Multithreaded Server(MTS) 结构的方法来多线程化不同种类的服务代理，这样就有一组调度程序来接收 Oracle数据库服务器进程的请求并把结果返回给它们。还有一个任务线程池来处理请求和计算结果。以这种方

法，可以调节大量的并发用户会话。虽然 Oracle进程的各个线程名不能从操作系统（Task Manager或Services窗口）中看到，但通过下列查询可以从SQL*Plus获得：

```
select b.Name bkpr, s.Username spid, p.Pid
  from V$BGPROCESS b, V$SESSION s, V$PROCESS p
 where p.Addr = b.Paddr(+)
    and p.Addr = s.Paddr;
```

BKPR	SPID	PID
-----	-----	-----
PMON		2
DBW0		3
LGWR		4
CKPT		5
SMON		6
RECO		7
SNP0		8
SNP1		9
	DBSNMP	12
	SYS	13

注意 这个列表不显示监听程序线程和调度程序线程。

15.2 Oracle和Net8

正如在第13章和14章所看到的，有许多不同的部件涉及到配置和使用 Net8。必须安装 Net8软件和配置各种不同的文件（listener.ora、tnsnames.ora和sqlnet.ora）。如果使用 Oracle Enterprise Manager（企业管理器）的 Intelligent Agent（智能代理），也需要配置 snmp_or.ora（该文件不能修改）和 snmp_rw.ora（该文件可以修改）。

当一台计算机与另一台计算机进行通信时，通过一组称为栈（stack）的特殊通信层来传递请求。栈的每个层执行一种不同的功能，把每个消息转换成一系列电脉冲传送给基本物理硬件。每种不同类型的网络专用协议都有所不同，但类似于栈，因为基本任务是使一台计算机的信息通过导线传到另一台计算机。

Net8是一个位于网络协议顶层的层栈。每台与数据库通信的机器都必须安装有 Net8。表15-2列出了适用于客户机的Net8层。

表15-2 Net8客户机组件

栈 层	描 述
Oracle Call Interface(OCI)	由许多功能调用组成。这些功能调用可嵌入程序中以便允许发送和接收往来于数据库的调用
The Two-Task Common	消除发送端和接收端之间的字符集差别
Net8（由3个层组成）	Network Interface(NI)对客户机应用程序隐藏基本网络协议和介质 Network Routing(NR)/Network Naming(NN)/Network Authentication(NA) 负责发送数据到终端目的地
Oracle Protocol Adapter	Transparent Network Substrate(TNS)负责一般通信，像发送和接收数据
网络协议	使Net8和基本网络协议隔离的瘦代码层 通常是另一个层栈，它把SQL语句转换成包以便在网络间发送

在服务器端，除了顶层以外，Net8的层是相同的。没有 Oracle Call Interface(调用接口)而有一个 Oracle-side Programmatic Interface(OPI, Oracle端程序设计接口)。如果正在使用 Internet

Inter-ORB Protocol(IIOP)或Oracle的Java Database Connectivity(JDBC)驱动程序,那么协议栈就与表15-2所示稍有不同。

15.2.1 Net8监听程序

NT机器中的Net8监听程序是一种服务,通常当引导机器时,就自动启动该服务。有三种不同的方法可手工启动监听程序:

1) 从Services窗口中选择 Oracle<oracle_home>TNSListener选项并选中Start按钮。

2) 从Start | Programs菜单中选择DOSwindow选项。在DOS窗口发出命令:

```
lsnrctl start<listener_name> (缺省为LISTENER)
```

3) 可以创建一个.bat文件并在该文件中发出命令:

```
net start Oracle<oracle_home>TNSListener
```

一旦启动Net8监听程序,它就开始监听来自网络上客户机的连接请求。一旦接收请求,Net8就负责进行客户机和数据库之间的连接。根据Net8的配置情况,将按如下方式进行连接:

- 在全部完成客户机的处理过程之前,作为该客户机的排他使用的一个专用线程。
- 把连接传送到预产生的线程。
- 通过Multithreaded Server(MTS,多线程服务器)或Connection Manager(CM,连接管理器)。

一旦连接被从监听程序传送到线程,就完成了监听程序的处理,并且在监听程序和该特定客户机的连接之间不再有进一步的交互。如果由于某种原因必须关闭和重新启动监听程序,那么当前的客户机到数据库的连接不受这些动作的影响。

15.2.2 使用多线程服务器

虽然多线程服务器在其他平台上已经使用了好多年了,但由于Oracle8.0版带有NT版本4.0,它只能在NT上使用。在NT版本4.0,Windows套接字版本2(WINSOCKET2)已被修改为包含更多的网络功能。现在调度程序和共享服务器已作为轻量线程被实现,因而整个系统变得更快更有效。但是请注意,MTS只使用TCP/IP进行操作,不支持其他协议。在客户机具有某些涉及其处理过程(如决策支持系统和联机分析处理)的“思考”时间的环境中,MTS会工作得更好。大型批装载环境或具有快速数据输入的环境不适合于NTS。

要配置NT上的MTS,可在init.ora文件中设置MTS参数值。诸如MTS_MAN_DISPATCHERS和MTS_MAX_SERVERS等参数可告知Oracle你想启动多少个调度程序线程和共享服务器线程。这些线程允许通过少量的物理上的数据库连接来支持大量的用户。

当使用MTS时,启动监听程序和数据库的顺序会变得很关键,必须以特定顺序执行下列任务:

1) 首先必须启动监听程序。

2) 启动监听程序之后,启动数据库。

3) 根据已在init.ora文件中设置的配置参数,启动调度程序并开始监听已分配的地址。

每个调度程序都向Net8监听程序来注册它的地址,缺省的监听程序地址是1521和1526。如果监听程序不监听缺省地址,那么可用LOCAL_LISTENER参数把监听程序的网路名放入

init.ora文件，以便以下列形式声明一个或多个监听程序：

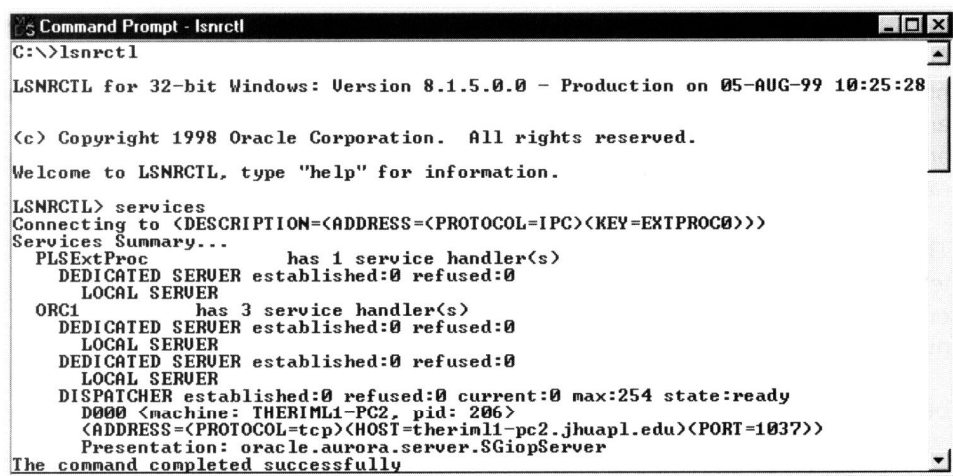
```
local_listener = "(ADDRESS_LIST = (Address = (Protocol = TCP) (Host=HQ1)  
(Port=1524)) (Address=(Protocol = IPC) (Key= HQ)))"
```

由于LOCAL_LISTENER参数是一个串值，因此要使用引号。在这个例子中，监听程序位于HQ1机并监听端口1524（非缺省端口号）。key值（HQ）是数据库名称。监听程序和数据库必须在同一台计算机上使用init.ora文件中的LOCAL_LISTENER参数。

在调度程序向监听程序注册后，就可通过监听程序来进行客户机和调度程序之间的连接。要查看已向监听程序注册的调度程序，可从DOS命令行发出下列命令：

```
lsnrctl  
LSNRCTL> services
```

图15-4示出了具有该命令结果的DOS命令窗口。在本例中，只有指定的服务器进程可用，指定的连接是Net8缺省。



```
Command Prompt - lsnrctl  
C:\>lsnrctl  
LSNRCTL for 32-bit Windows: Version 8.1.5.0.0 - Production on 05-AUG-99 10:25:28  
  
(c) Copyright 1998 Oracle Corporation. All rights reserved.  
Welcome to LSNRCTL, type "help" for information.  
  
LSNRCTL> services  
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC)))  
Services Summary...  
  PLSExtProc      has 1 service handler(s)  
    DEDICATED SERVER established:0 refused:0  
    LOCAL SERVER  
  ORCL            has 3 service handler(s)  
    DEDICATED SERVER established:0 refused:0  
    LOCAL SERVER  
    DEDICATED SERVER established:0 refused:0  
    LOCAL SERVER  
  DISPATCHER established:0 refused:0 current:0 max:254 state:ready  
  D000 <machine: THERIML1-PC2, pid: 206>  
    (ADDRESS=(PROTOCOL=tcp)(HOST=theriml1-pc2.jhuapl.edu)(PORT=1037))  
  Presentation: oracle.aurora.server.SGiopServer  
The command completed successfully
```

图15-4 具有命令结果的DOS命令窗口

可以使用Oracle Net8Configuration Assistant和Net8 Easy Config工具来配置listener.ora文件和tnsnames.ora文件。有关如何使用这两个工具的详细情况，请参见第13章。

15.3 配置Windows NT为后端服务器

在前面几节中，已经介绍了NT上支持一般客户机/服务器的联网能力和配置选项。本节将介绍如何把NT机配置为更高效的数据库服务器。

在一个NT系统中，有4GB的潜在内存空间可以寻址。这些可寻址空间的一半专用于系统服务和文件高速缓存，诸如Oracle8i这样的用户程序则使用这些可用空间的另一半。由于大多数Windows NT系统实际上没有真正安装4GB内存，所以这些系统都采用一种称为“虚拟内存”的机制来平衡内存的使用。Windows NT系统使用一个虚拟内存分页文件来动态交换物理RAM和虚拟分页区域之间的内存页面。Oracle8i是一种内存密集型应用，所以内存需求可能会超过机器中的实际内存容量。

有几个步骤可以用来把Windows NT配置为后端服务器。第一个步骤是必须保证这台机器不能用于其他任何活动，如主域或备份域控制器、文件或打印服务器、远程访问服务器或者路由器或防火墙服务器。在下面的各小节中，将详细介绍可用来优化NT服务器性能的其他步骤。

15.3.1 降低交互式前台应用程序的优先权

接下来的操作就是降低服务器控制台上的交互式前台应用程序的优先权，因为已给予它们后台进程的优先权。要降低交互式前台应用程序的优先权并因此增加可用于 Oracle8i 的资源，可执行下列步骤：

- 1) 从 Settings 菜单下的 Control Panel 中选择 System 选项。
- 2) 单击 Performance 选项卡。
- 3) 把 Application Performance Boost 滑块移到 None。
- 4) 确保虚拟内存分页文件的大小至少设置为 300MB。
- 5) 单击对话框中的 OK。
- 6) 退出 Control Panel。

如果在系统中安装了多个物理磁盘，则虚拟内存分页文件可跨这些物理卷。如果系统中具有逻辑的而不是物理的磁盘，则跨这些磁盘放置虚拟内存分页文件是不值得的。如果改变了虚拟内存分页文件的大小，就必须重新启动系统以使这种改变生效。

15.3.2 减少 Windows NT 服务器的文件缓存

通常，Windows NT 保留了 41% 的可用内存用于系统中的文件缓存活动。虽然这对于作为服务器使用的计算机来说能很好地利用资源，但对于 Oracle8i 数据库，这却是不必要的，Oracle8i 数据库通过 SGA 进行自己的缓存。减少文件缓存分配的目的是为 Oracle8i 应用程序提供更多的可用内存。要减少文件缓存，可执行如下步骤：

- 1) 从 Settings 菜单下的 Control Panel 中选择 Network 选项。
- 2) 单击 Services 选项卡。
- 3) 选择 Server 服务并单击 Properties 选项。
- 4) 选择网络应用程序配置的按钮并单击 OK。
- 5) 单击 Network 对话框中的 OK。
- 6) 退出 Control Panel。

为了使新设置生效发挥作用，必须重新启动系统。

15.3.3 禁止不必要的服务

要恢复 Oracle8i 数据库的附加内存，可禁止核心操作系统功能或 Oracle8i 所不需要的服务。请注意，禁止不必要服务的建议是基于这样的假设，即 Windows NT 除用于 Oracle8i 之外不用于任何其他功能。如果是这种情况，可禁止下列服务： Plug and Play（即即用）、Remote Access Autodial Manager（远程访问自动拨号管理器）、Remote Access Connection Manager（远程访问连接管理器）、Remote Access Server（远程访问服务器）和 Telephony Service（通话服务）。要禁止这些服务，可执行如下步骤：

- 1) 从 Settings 菜单下的 Control Panel 中选择 Services 选项。
- 2) 依次选定上面所列的各个选项并选择 Startup 选项。
- 3) 在 Startup Type 下选择 Disable，然后单击 OK。
- 4) 重新设置每个选项后，单击 Close，退出 Control Panel。

15.3.4 取消不使用的网络协议并重新设置连接顺序

由于不是所有可在 Windows NT 机器上配置或分配的协议都被使用，许多本应由关键协议使用的处理时间可能被耗费掉。因此，你可能想取消所有未由后端服务器使用的网络协议。要取消不使用的网络协议，可先确定正在使用的协议，然后执行如下步骤：

- 1) 选择 Settings 菜单下的 Control Panel 中的 Network 选项。
- 2) 单击 Protocols 选项卡。
- 3) 依次选定每个不需要的协议并单击 Remove 选项。
- 4) 单击 Yes 以确认选择。
- 5) 一旦删除了所有的不需要协议，就单击 OK，退出。

必须重新启动机器以使新设置生效。

如果正使用多个协议，则可重新设置网络协议的连接顺序，以便给予 Oracle8i 数据库最经常使用的协议最高的优先权。要重新设置连接顺序，可执行如下步骤：

- 1) 选择 Settings 菜单下的 Control Panel 中的 Network 选项。
- 2) 单击 Bindings 选项卡。
- 3) 从 Show Bindings for 显示区域中选择 All Services。
- 4) 双击 Server 列表以扩展该列表。
- 5) 选择 Oracle8i 使用最多的协议（如果它没处于该列表的顶端的话）并单击 Move Up，直到它处于该列表的顶端。对所有安装在系统中的每个网络接口卡都可如此进行操作。
- 6) 单击 OK 并退出 Control Panel。

必须重新启动机器以使新设置生效。

15.3.5 其他可用的配置选项

虽然上面几节所列出的操作将把最多的资源交还给后端服务器，还可考虑采用其他的操作来帮助提供某些好处（尽管没有那么多）。

Microsoft 公司定期地发布一些带有修补程序的服务软件包以供故障修复和产品增强之用。在系统上应用任何新服务包之前，务必质询 Oracle 技术支持以确认该服务包确实可用于你的当前硬件配置。如果该服务包获得认可并且在该包中没发现什么主要毛病，那么应用这样的软件包有助于改善系统性能。

如果使用 Windows NT 系统作为后端服务器，则可关闭启动文件夹中的不必要的前台应用程序。对于不需要的应用程序，请查看 Winnt\Profiles\All Users\Start Menu\Programs\Startup 目录。像 FindFast 索引和 MS Office Shortcut Toolbar 这样的程序可被删除。

复杂的屏幕保护程序会快速使 CPU 饱和。如果需要把屏幕保护程序用于长时间运行的进程，则应使用空白或有限文本的屏幕保护程序以减少 CPU 冲击。通过删除 Oracle 不需要的程序和特性，可以改善服务器中数据库或数据库网络的性能。

作为一种附加的 NT 调整选项，可以保证所有的 SGA 页面都保持在内存中。要使该选项生效，应在启动数据库之前把 init.ora 参数 PRE_PAGE_SGA 设置为 TRUE（真）。只有当 NT 服务器有足够的内存来把整个 SGA 保存在内存中时才能使用该选项。使用 PRE_PAGE_SGA 选项将影响到数据库启动和用户登录所需的时间，但是涉及 SGA 的其他数据库活动的性能将得到改善。