第 9 章：监控和调整数据库

目的

本章向您介绍通过企业管理器执行的一些监控和调整操作。

主题

本章讨论了以下内容：

主动监控数据库

- 诊断和解决性能问题
- 使用 SQL Tuning Advisor
- 使用 SQL Access Advisor
- 使用 Memory Advisor

主动监控数据库

警报帮助您主动地监控数据库。 大多数警报是在超过特定的量度阈值时发出的通告。 您可以为每一个警报设置临界和警告阈值。 这些阈值即为边界值，如果超过了这些边界值，则指示系统处于非预期的状态。
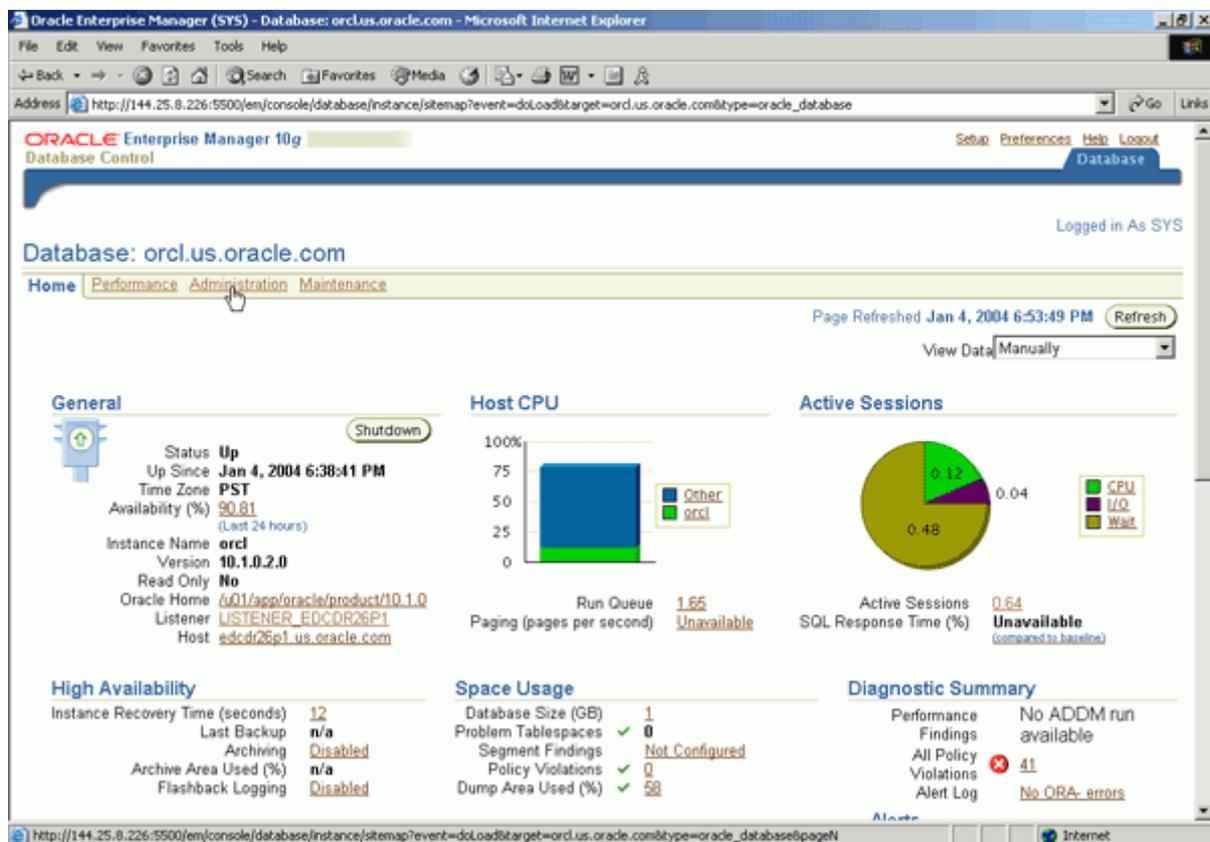
在这一部分中，您将执行以下任务：

- 创建具有指定阈值的表空间和表
- 触发表空间的空间使用率警报
- 设置量度阈值
- 设置通告
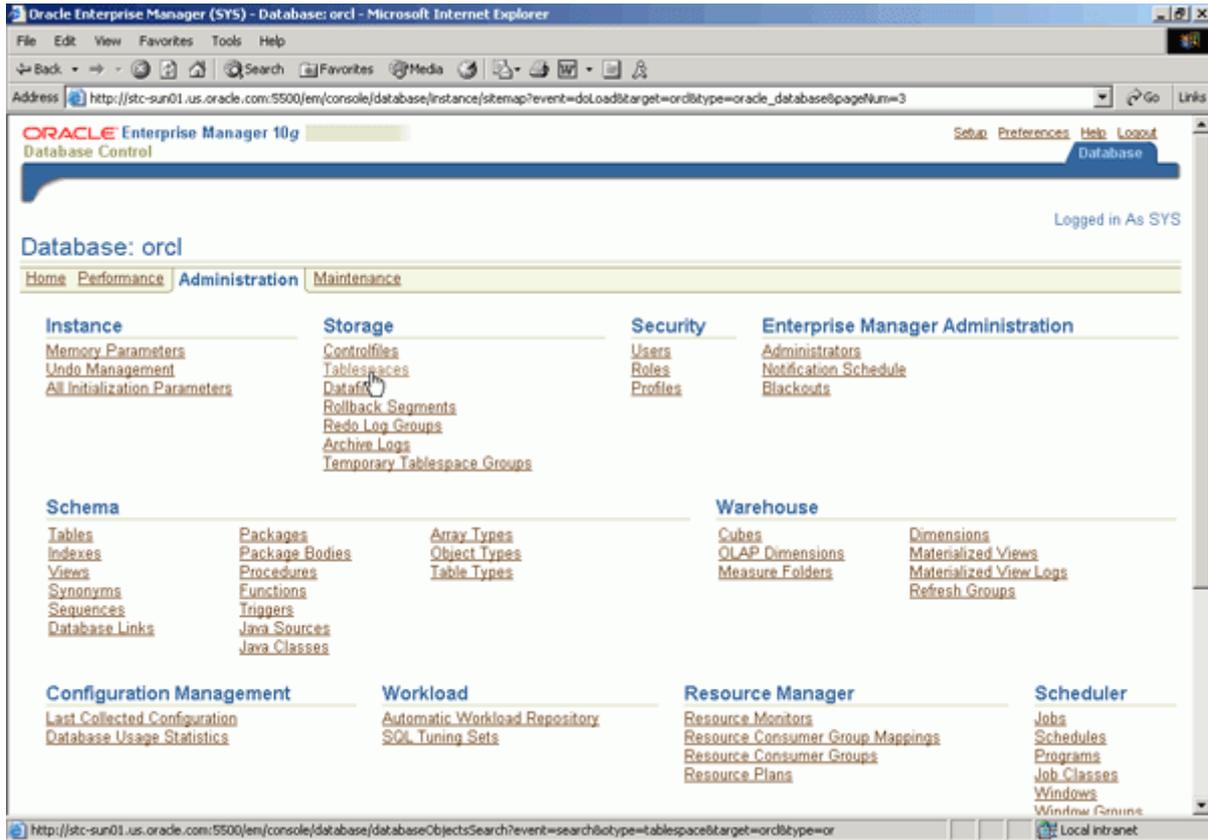
创建具有指定阈值的表空间和表

首先创建包含有一个 20 MB 的数据文件的一个新的表空间。 将本地管理这个表空间，并使用自动段空间管理 (ASSM)。 然后您在这个新的表空间中创建一个新的表。 这个表将把 Enable Row Movement 选项设为 yes，以允许表中的空间回收。 执行以下操作：
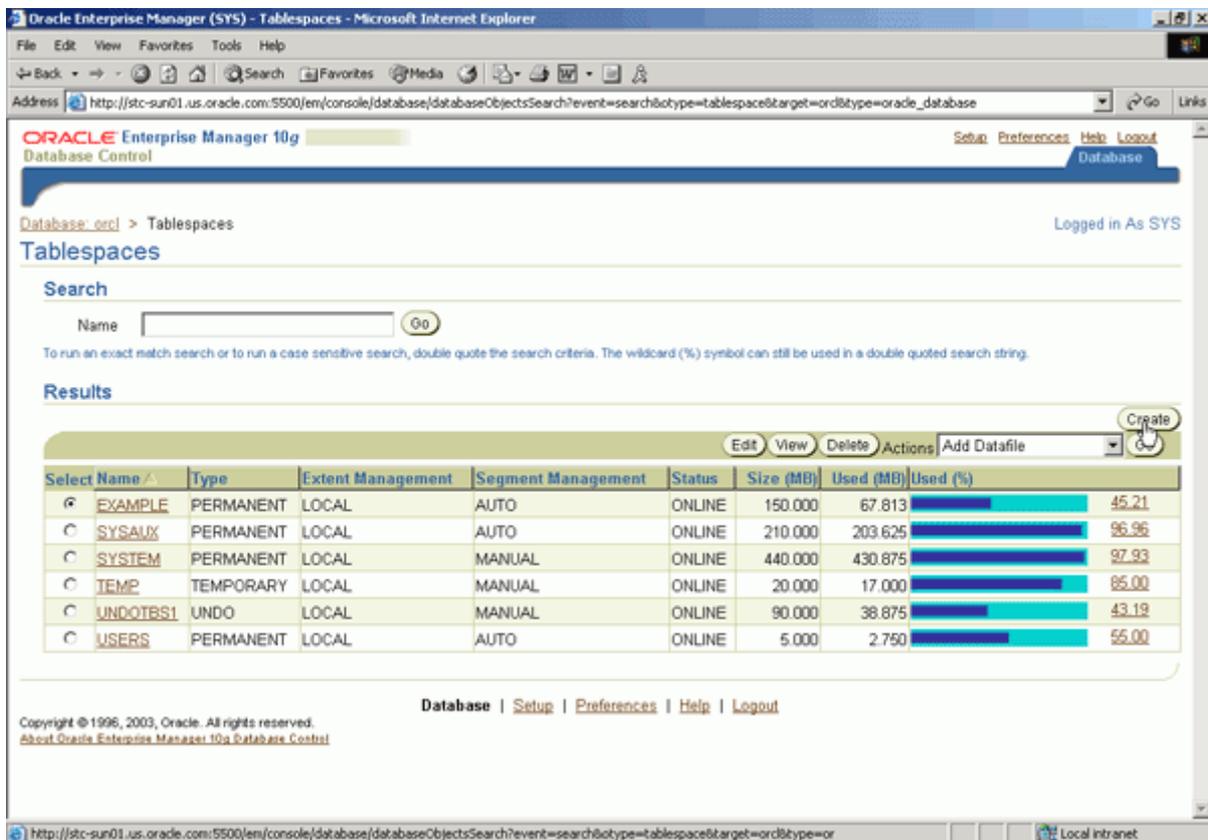
1. 单击数据库主页上的 Administration 链接。
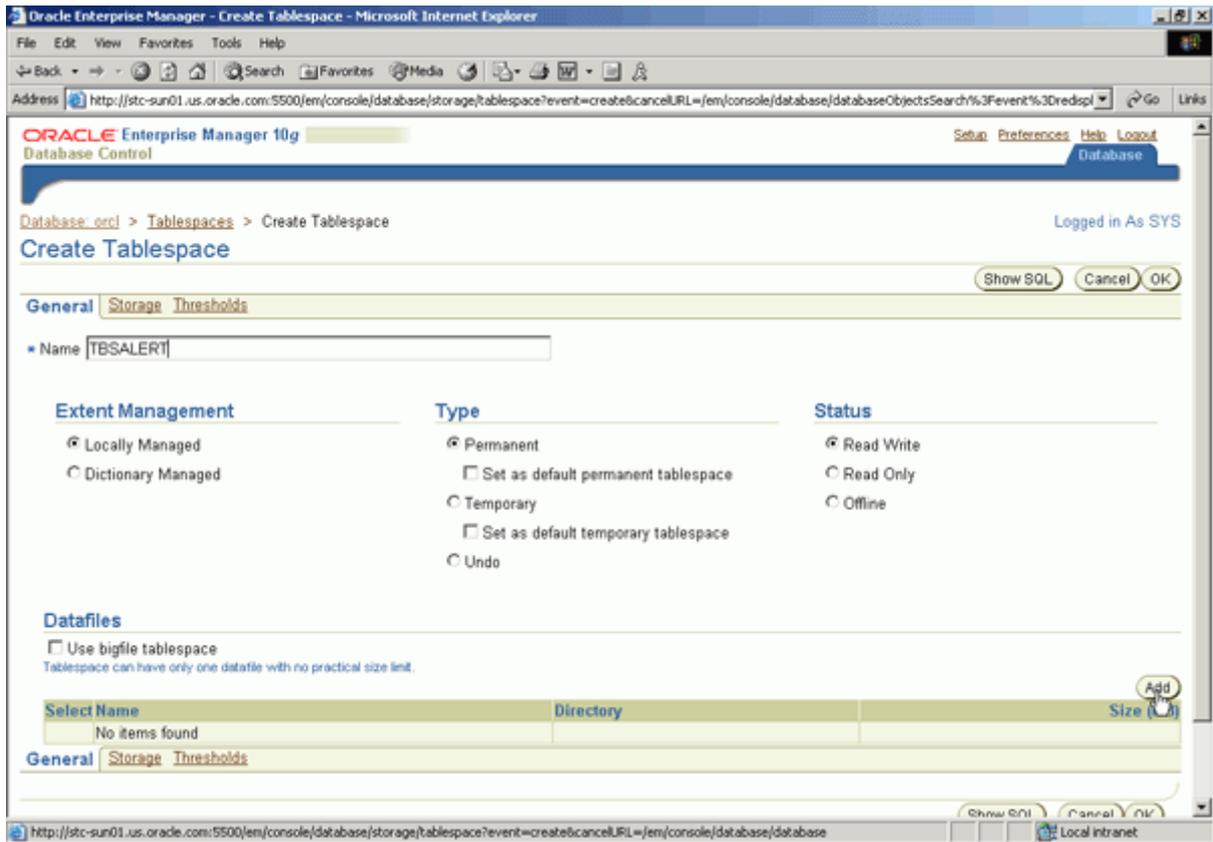
2. 单击 Tablespaces 链接。
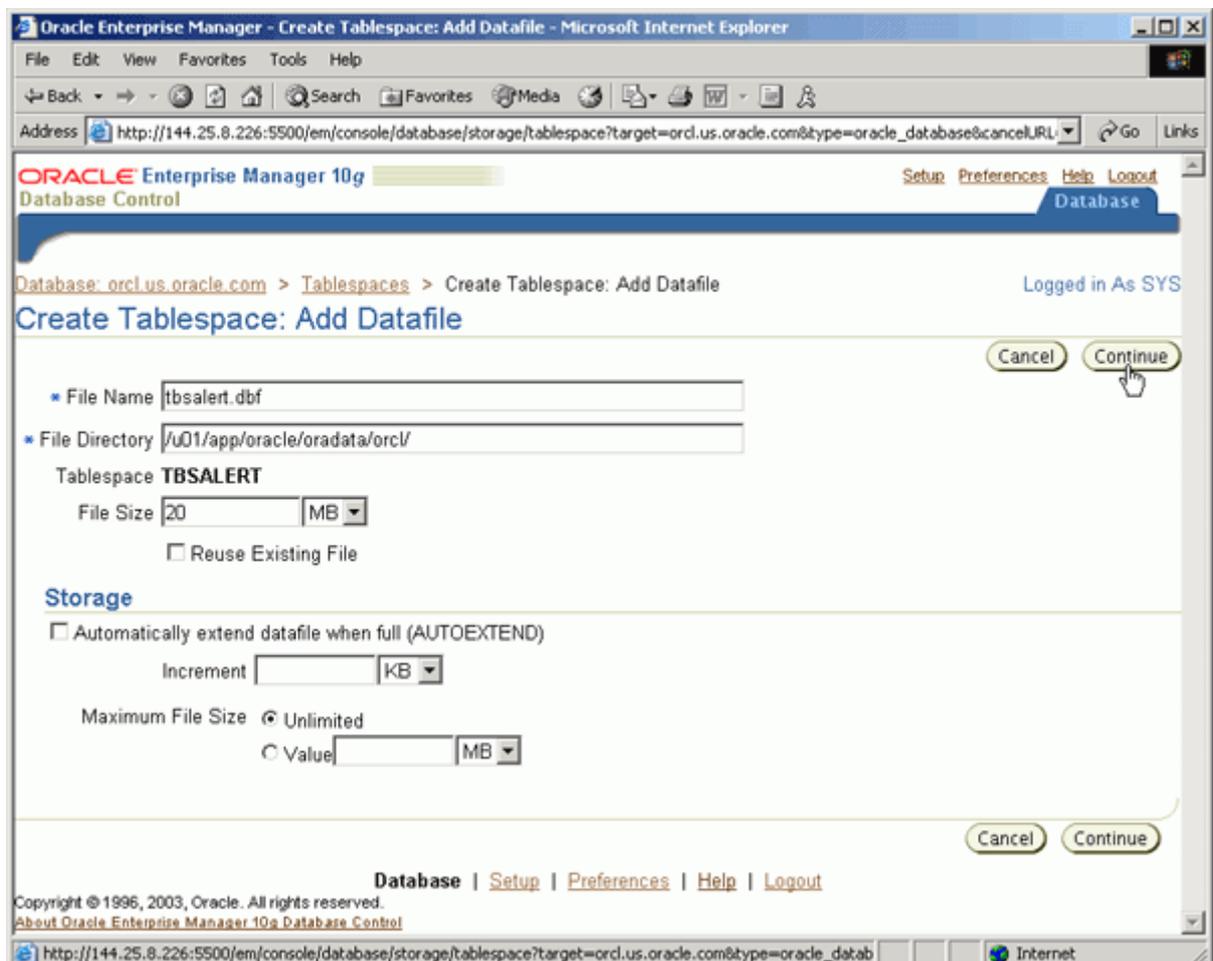
3. 单击 Create 按钮。
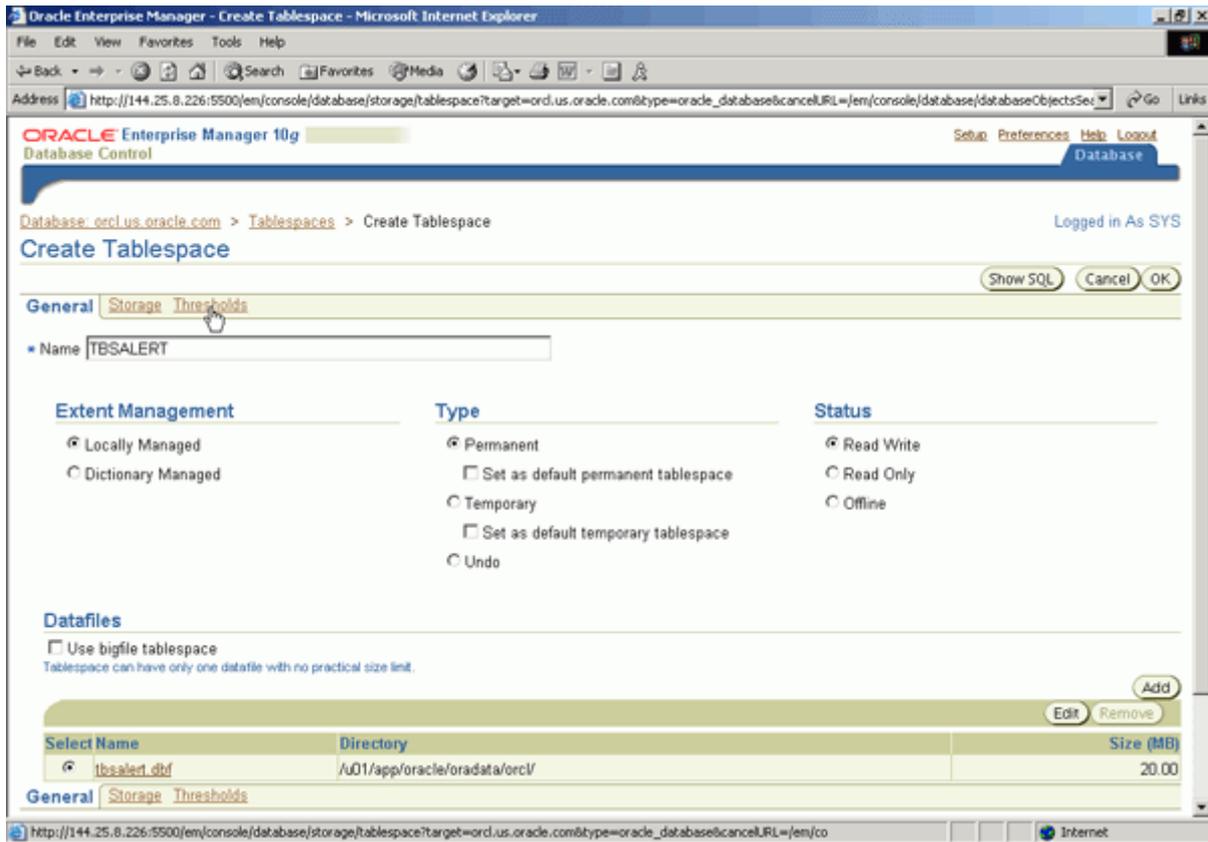
4. 输入 TBSALERT 作为表空间名称，然后单击 Add ，为表空间定义一个数据文件。
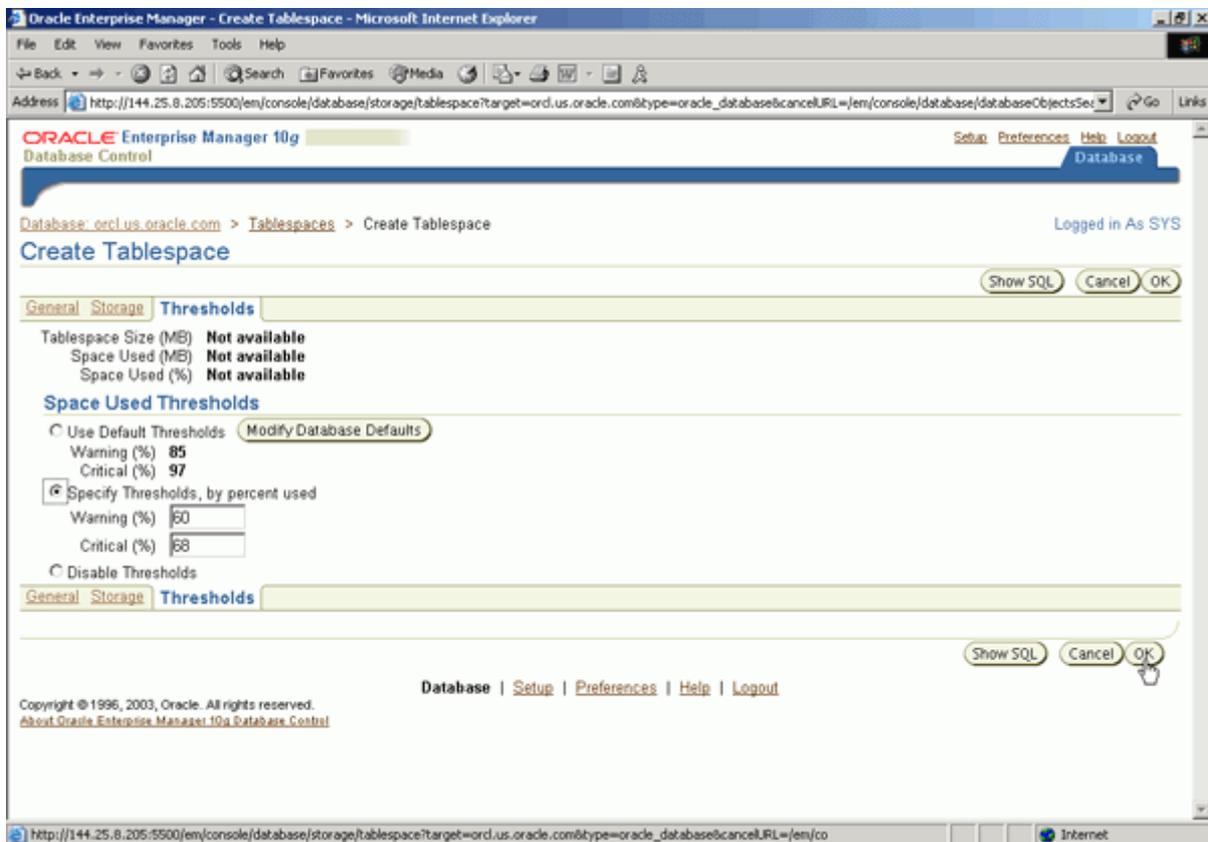
5. 输入 tbsalert.dbf 作为数据文件名，并输入 20 MB 作为文件大小。 单击 Continue 。
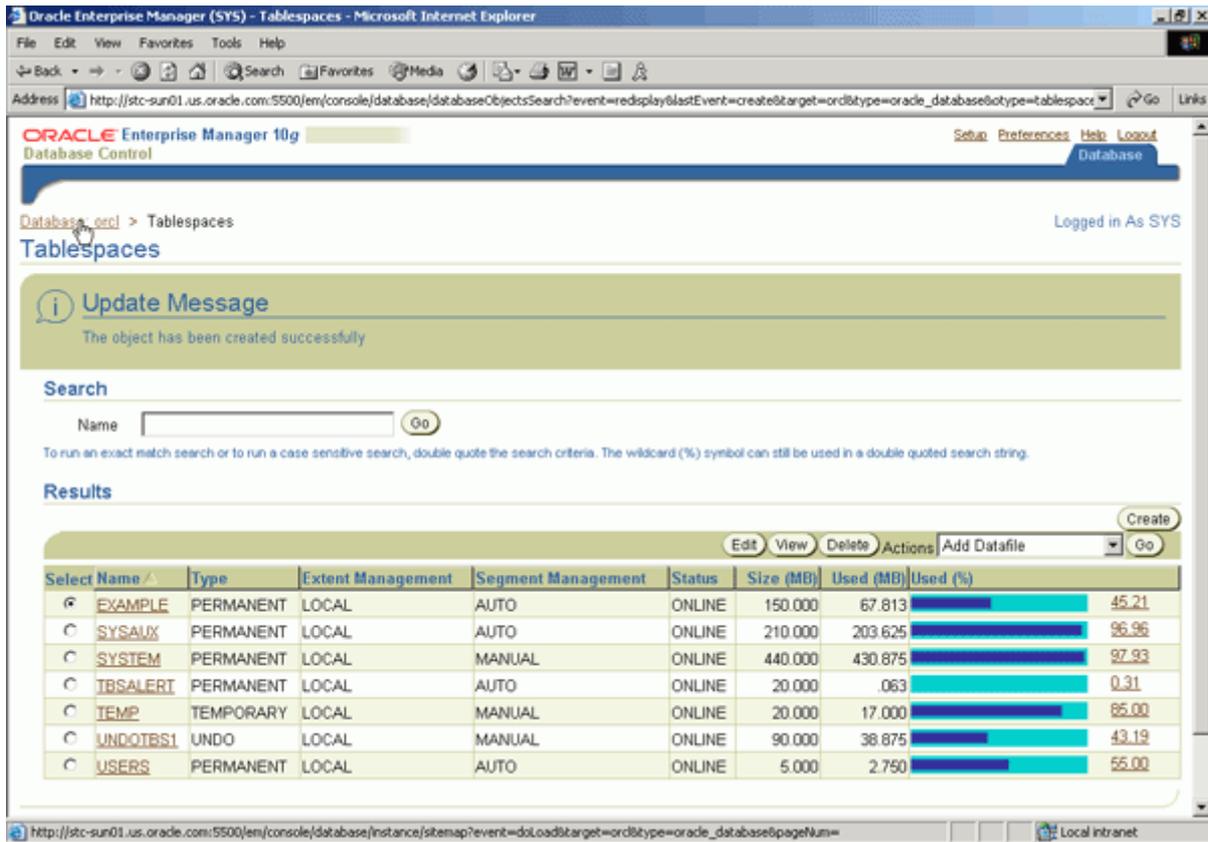
6. 单击 Thresholds ，指定空间使用率警告和临界阈值水平。
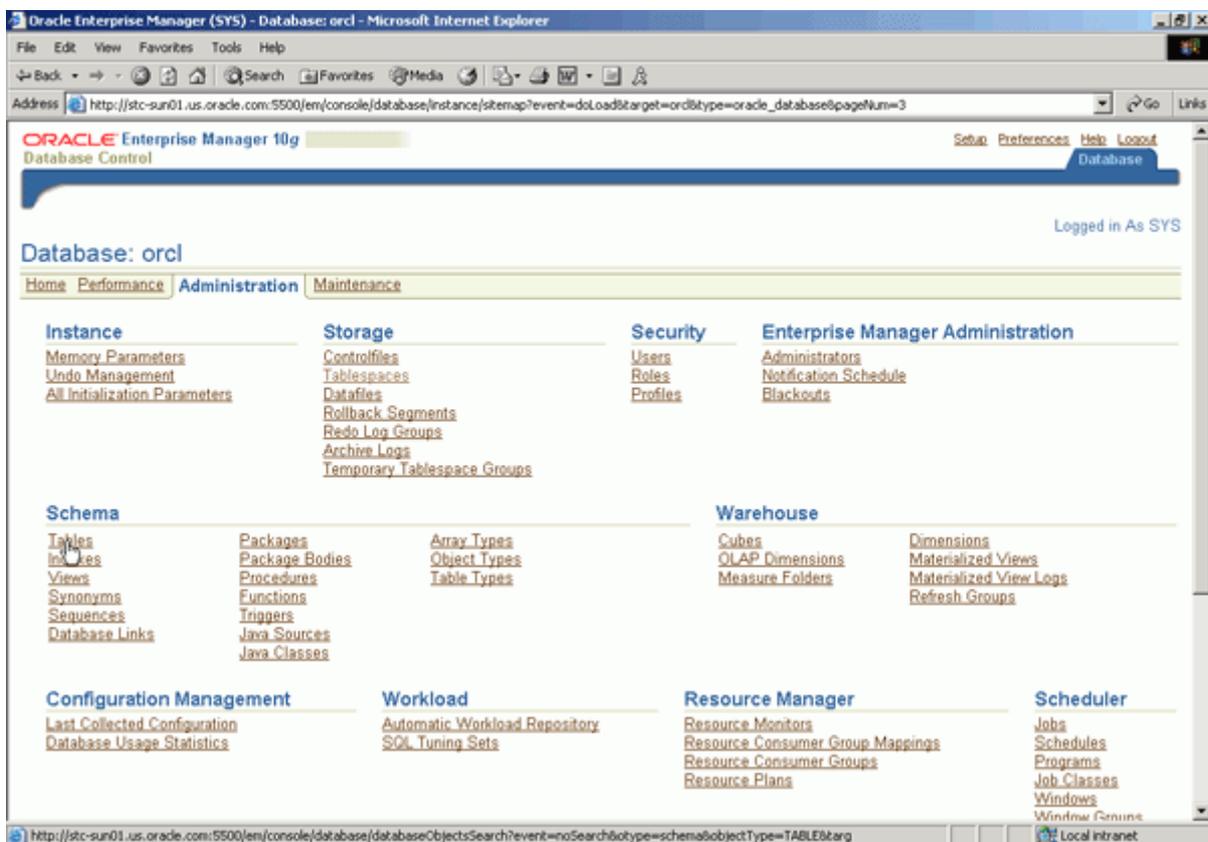
7. 单击 Specify Thresholds, by percent used 。 设置数据库范围的默认空间使用率阈值。 为 Warning% 输入 60 ，为 Critical% 输入 68 。单击 OK 。
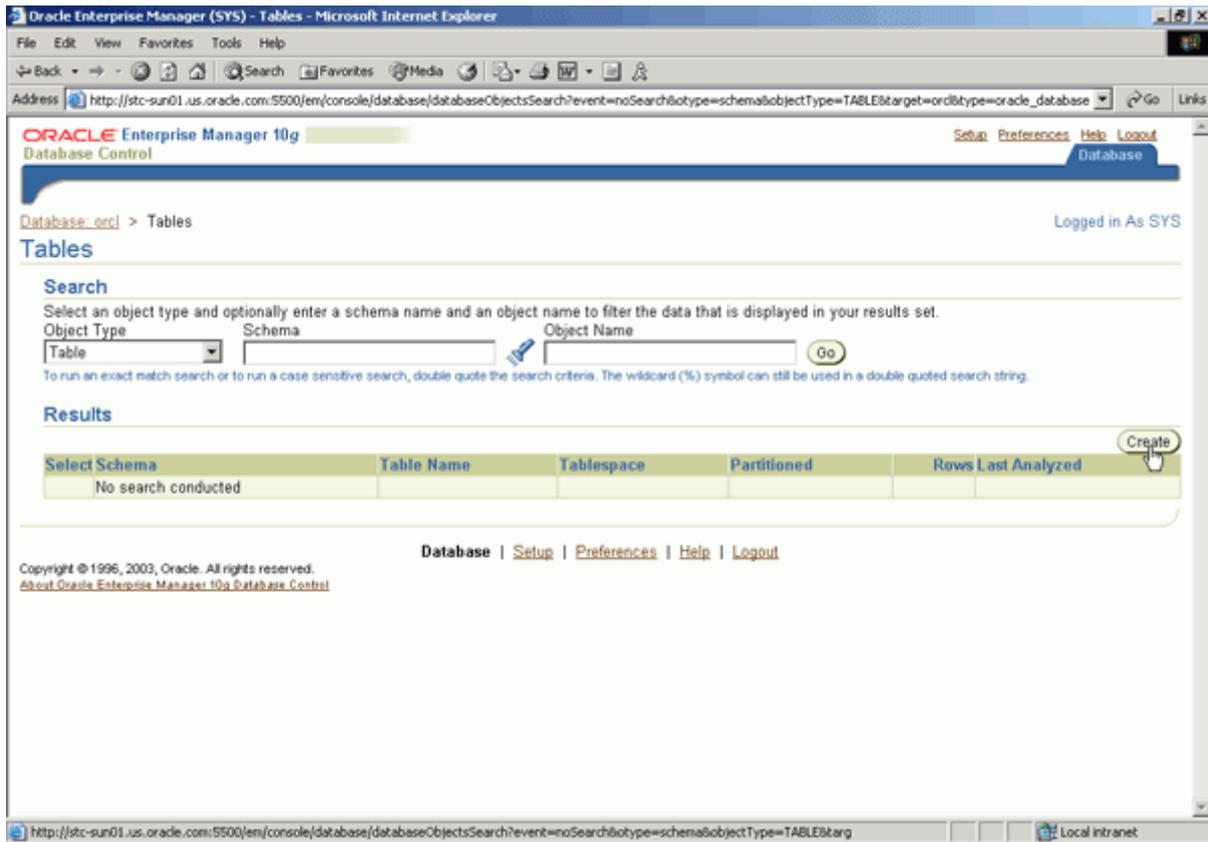
8. 单击 Database 导览栏，返回 Administration 属性页面。
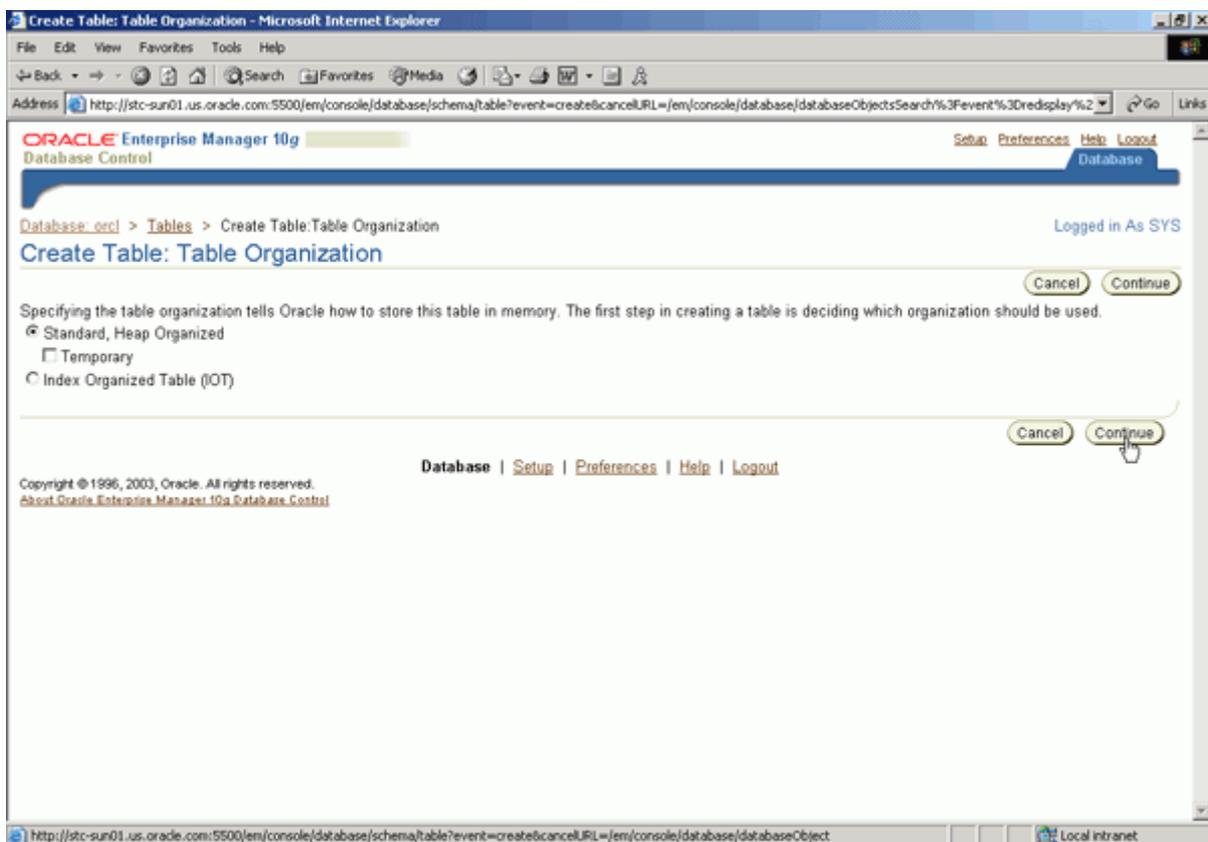
9. 单击 Tables 链接。
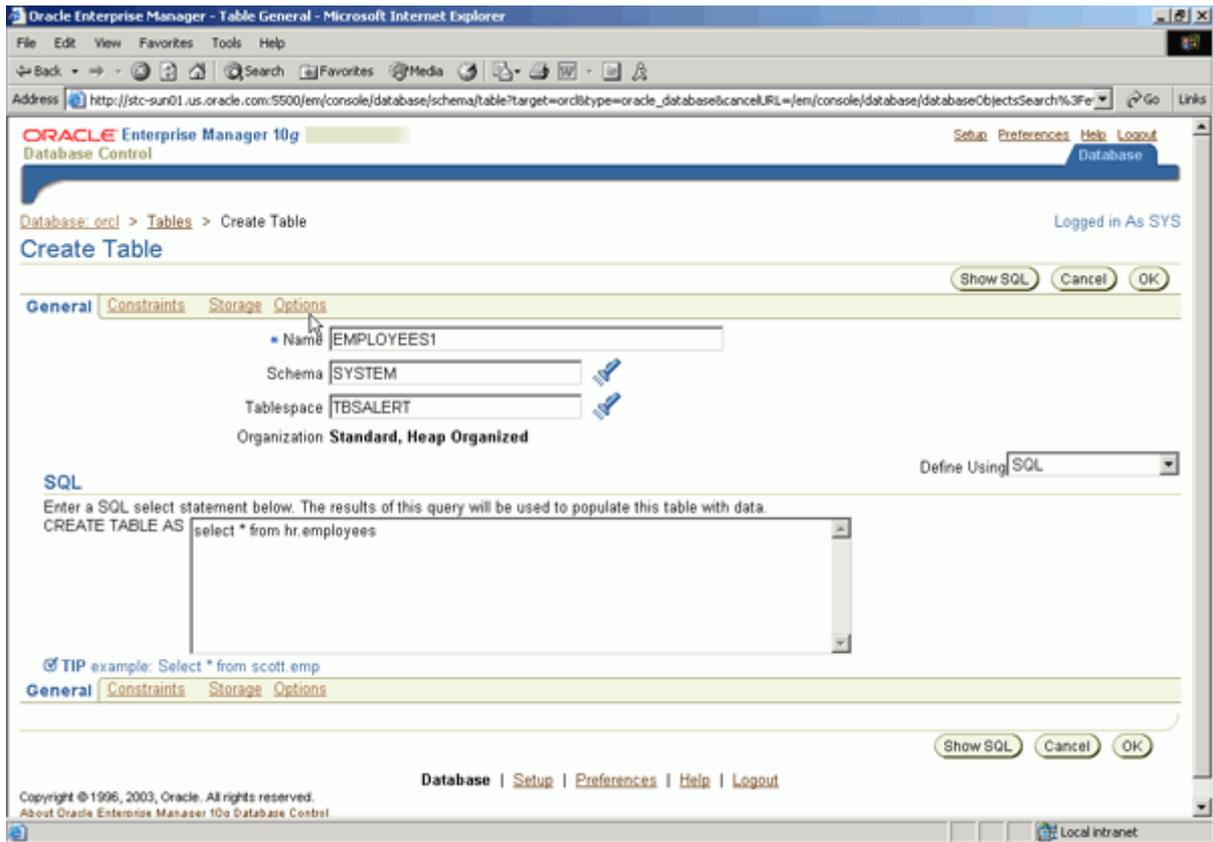
10. 单击 Create 。
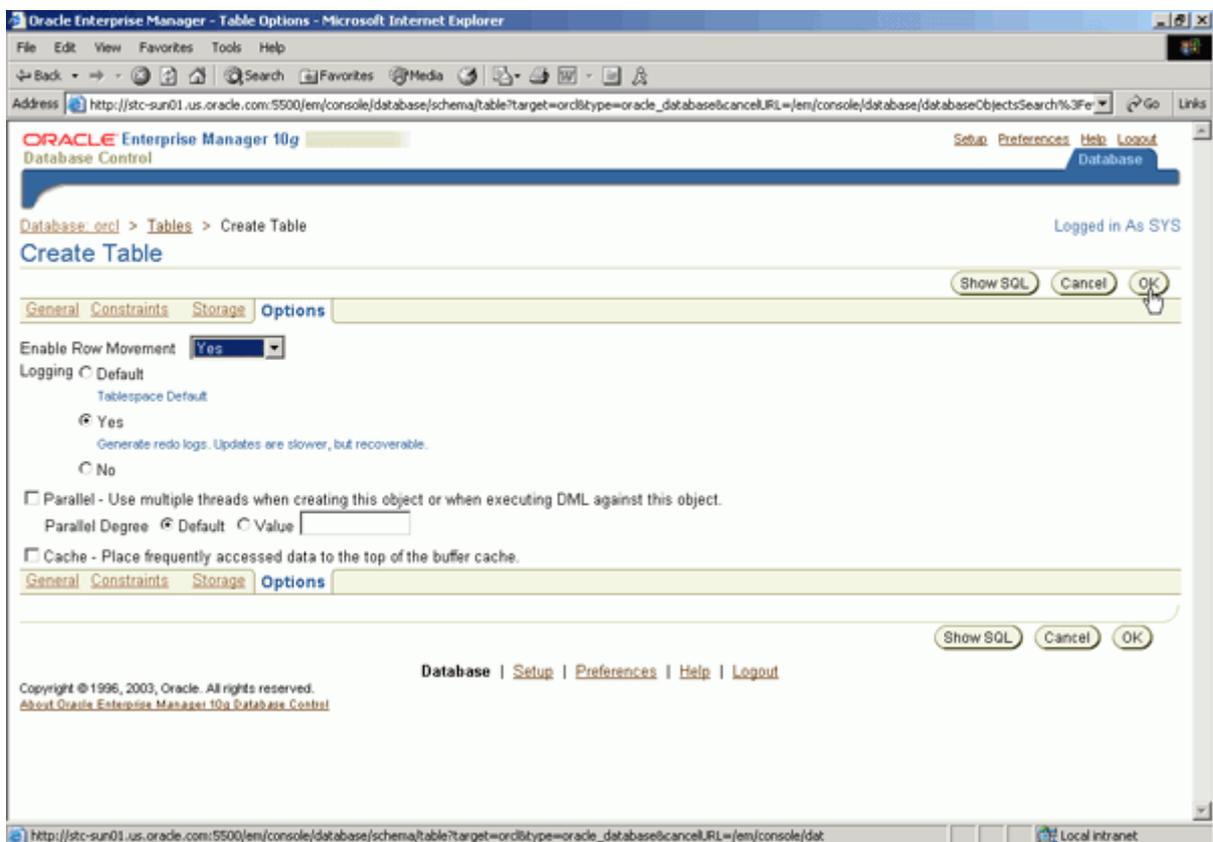
11. 单击 Continue 。
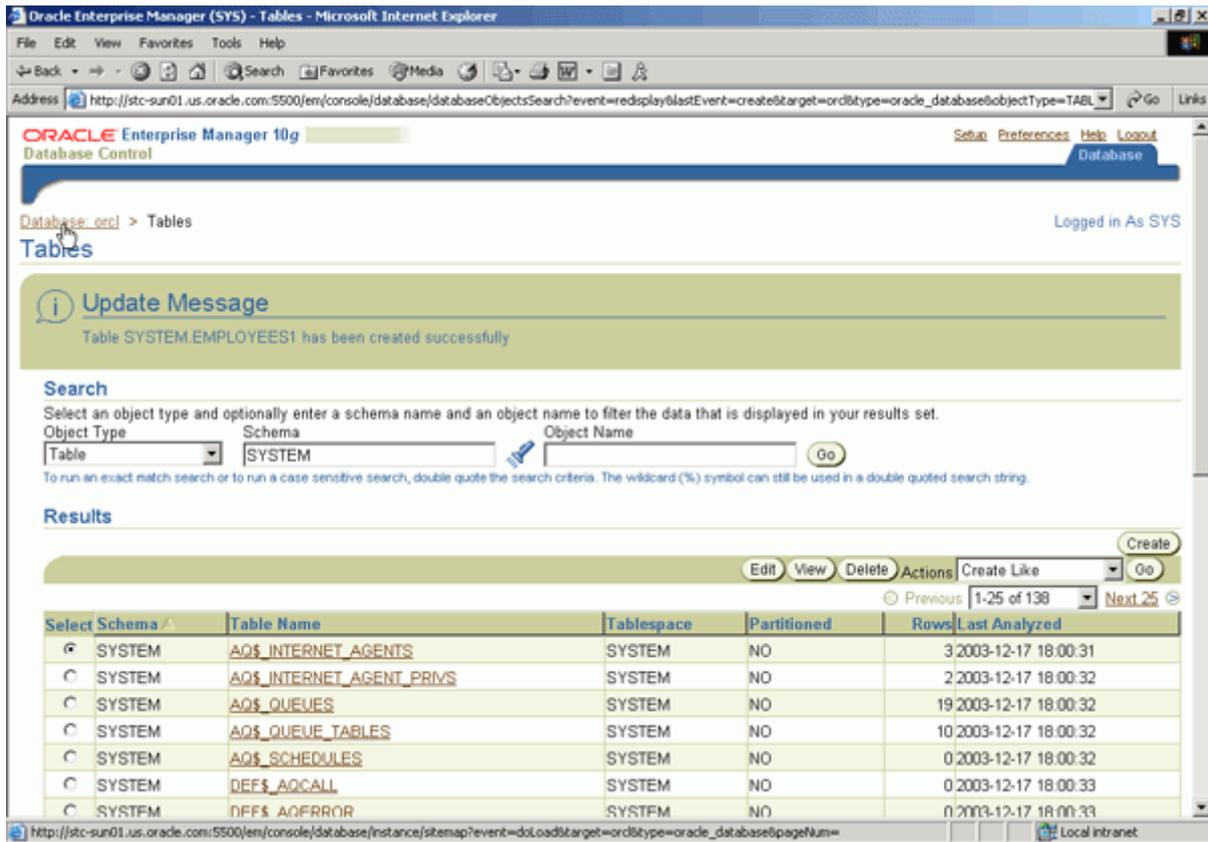
12. 在 Name 域中输入 employees1 。 指定 SYSTEM 作为模式， TBSALERT 作为表空间。 单击 Define Using 下拉列表，选择 SQL 。 在 Create Table As 域中输入 select * from hr.employees ，然后单击 Options 。

13. 为 Enable Row Movement 选择 Yes ，然后单击 OK ，完成表的创建。

14. 表已创建完成。 单击 Database 导览栏。

触发表空间空间使用率警报
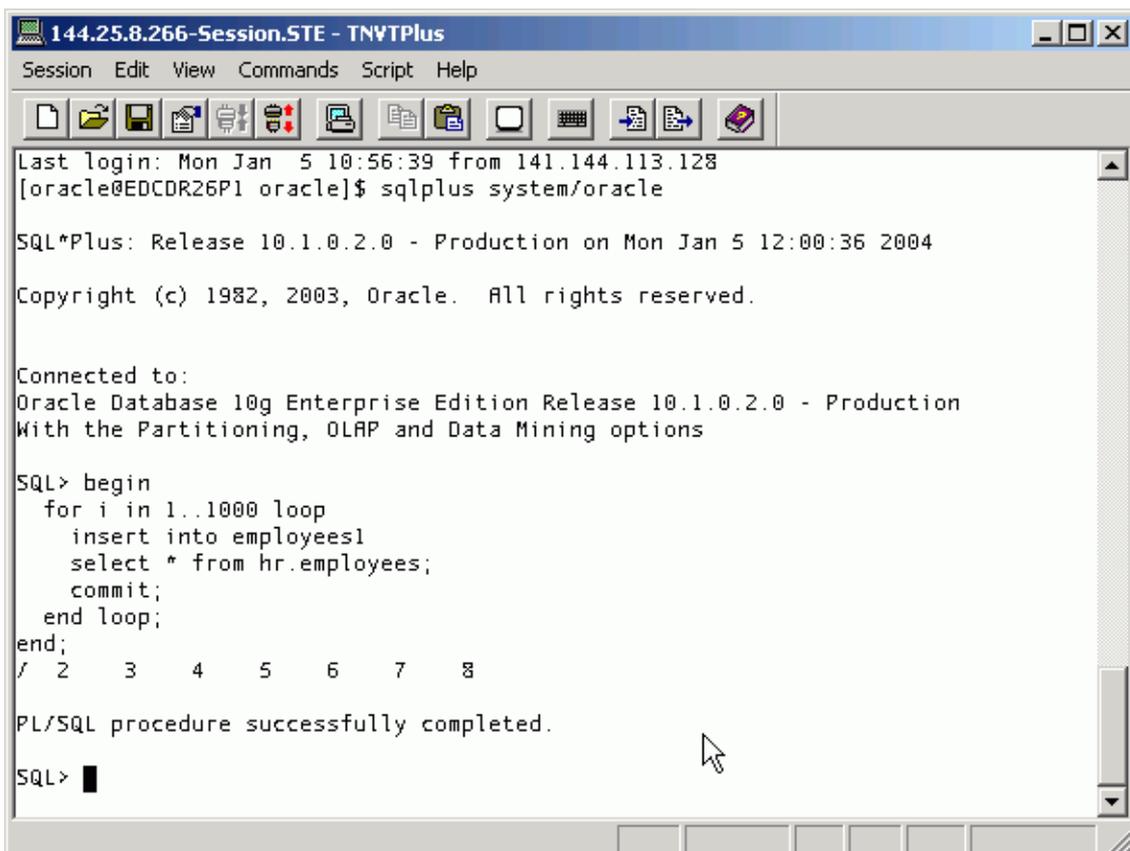
您现在将更新表，触发空间利用率警报。 执行以下操作：

1. 打开 SQL*Plus 会话，然后执行以下命令：
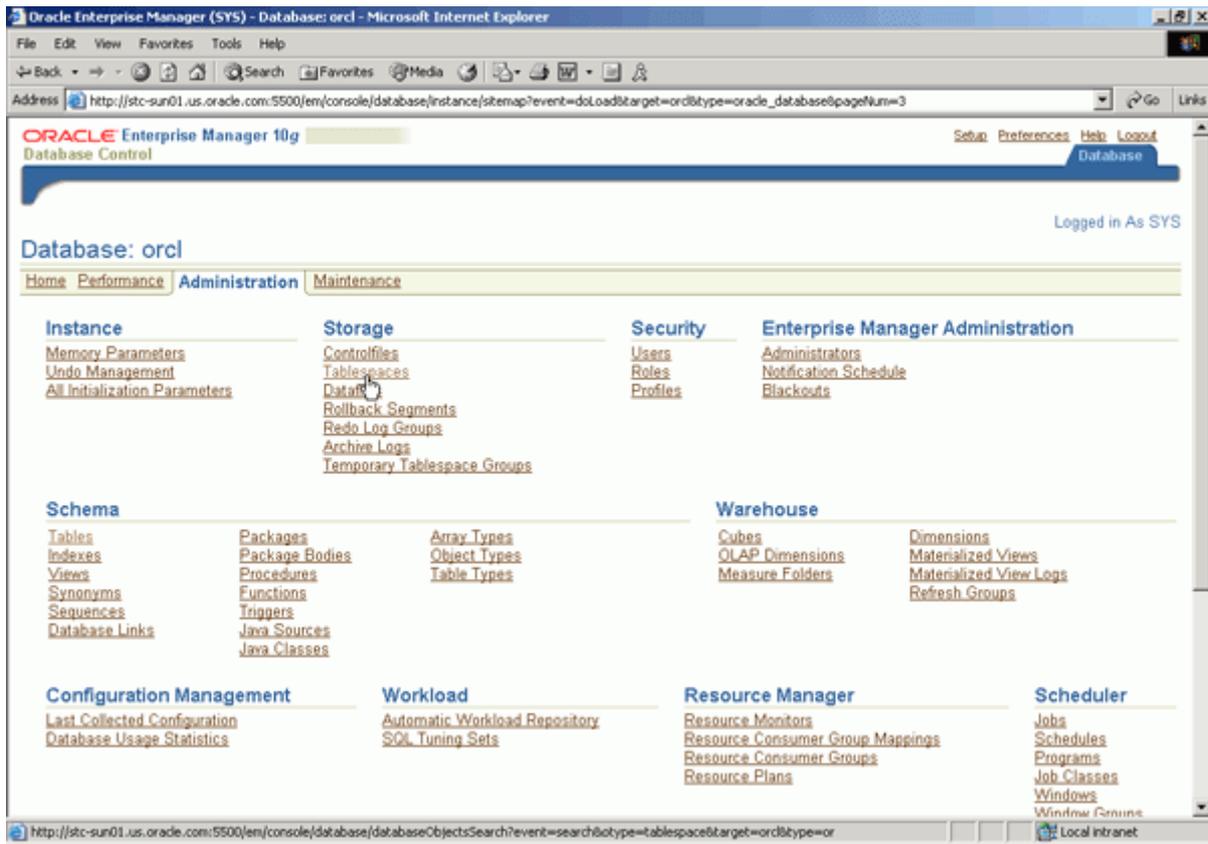
sqlplus system/oracle

2. 将以下 SQL 命令拷贝并粘贴到 SQL*Plus 会话中， 以模拟 EMPLOYEES1 表上的用户活动：
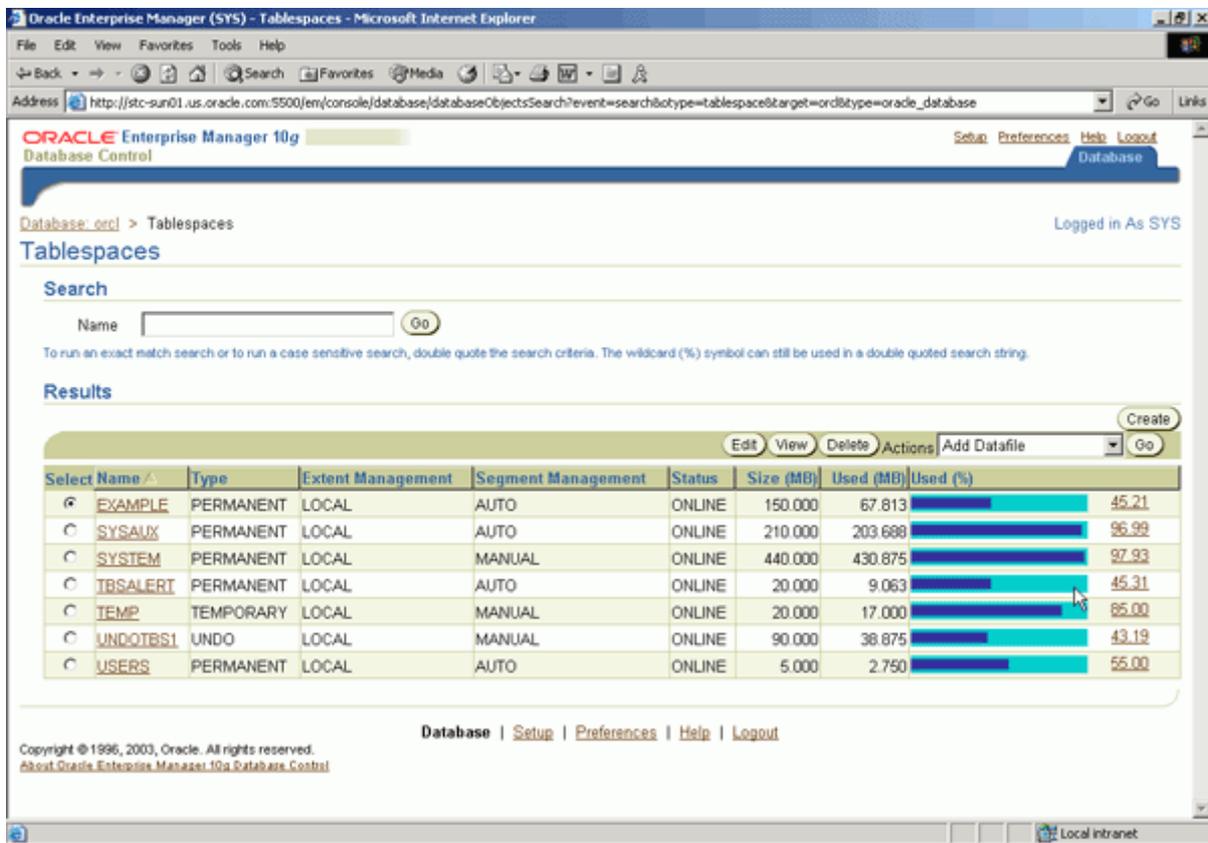
begin for i in 1..1000 loop insert into employees1 select * from hr.employees; commit; end loop; end; /

3. 转至 Enterprise Manager 窗口。单击 Database 导览栏,返回到 Administration 页面,然后单击 Tablespaces
链接。
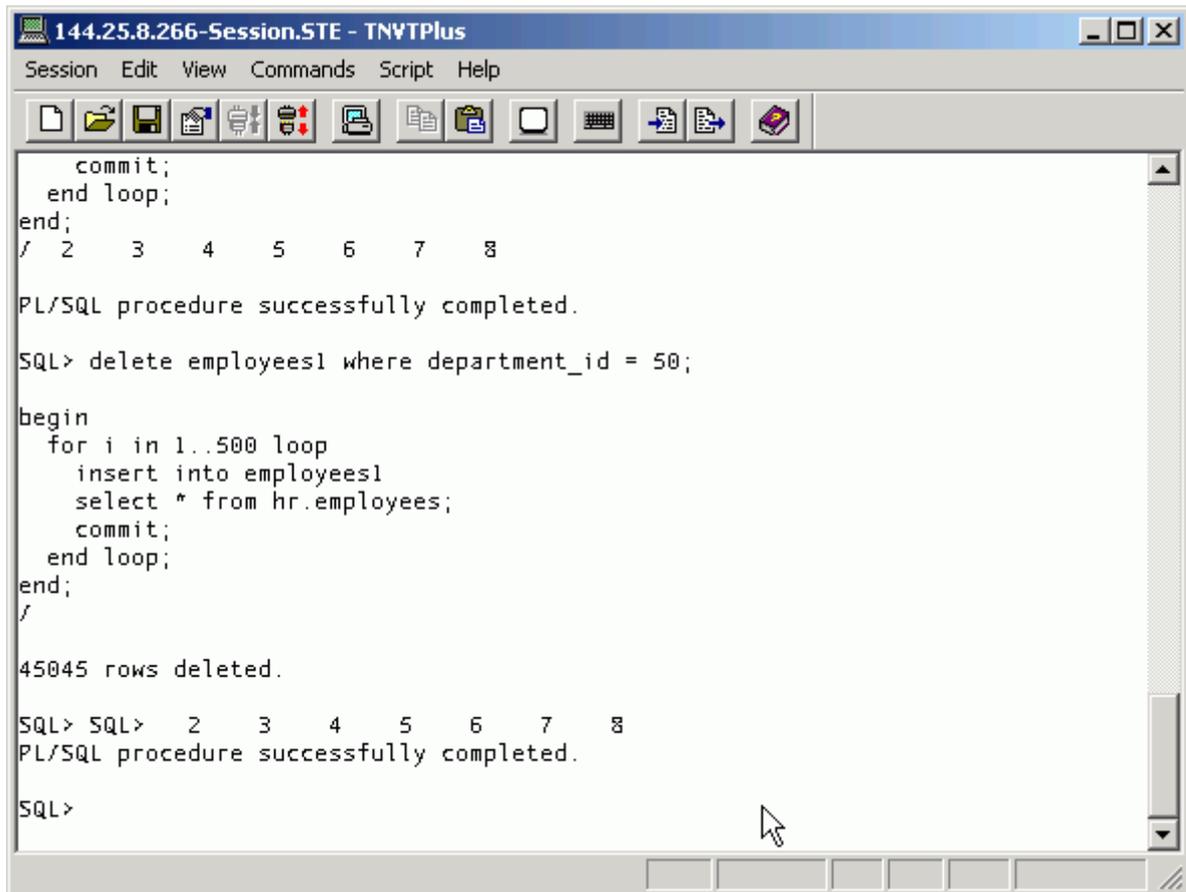
4. 注意, TBSALERT 表空间使用百分率增加了。

5. 回切到 SQL*Plus 窗口,并将以下命令拷贝并粘贴到 SQL*Plus 会话中, ,以模拟 EMPLOYEES1 表上的
更多用户活动:

delete employees1 where department_id = 50; begin for i in 1..500 loop insert into employees1 select * from hr.employees; commit; end loop; end; /
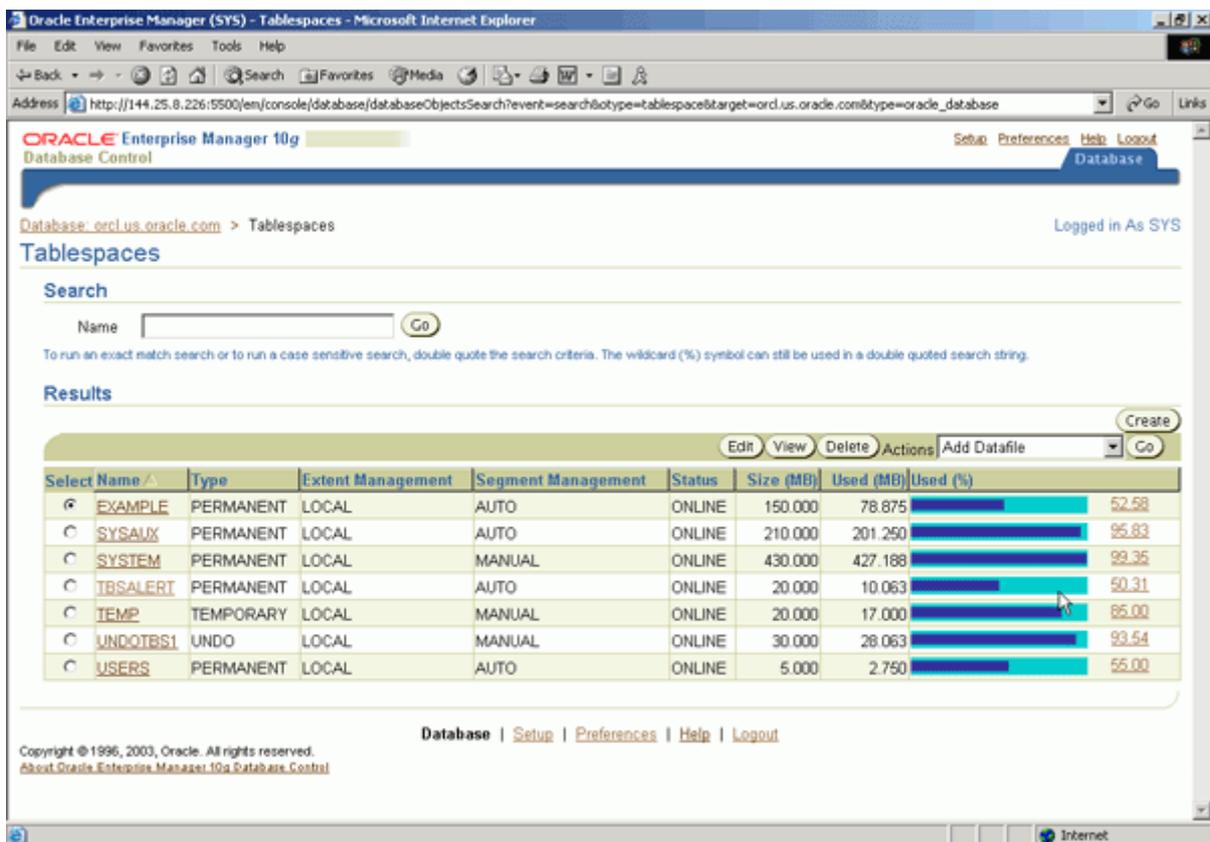
6.   转至 Enterprise Manager 窗口。 刷新浏览器（对于 Linux Mozilla，从菜单栏中选择 View ，然后选择 Reload ）。 注意， TBSALERT 表空间的空间使用百分率增加了。

7.   回切到 SQL*Plus 窗口，并将以下命令拷贝并粘贴到 SQL*Plus 会话中， ，以模拟 EMPLOYEES1 表上的

更多用户活动：

begin for i in 1..500 loop insert into employees1 select * from hr.employees; commit; end loop; end; /

8. 将以下 SQL 命令拷贝并粘贴到 SQL*Plus 会话中， 以模拟 EMPLOYEES1 表上的用户活动：

delete employees1 where department_id = 30; commit; delete employees1 where department_id = 100; commit; delete employees1 where department_id = 50; commit; delete employees1 where department_id = 80; commit; exit
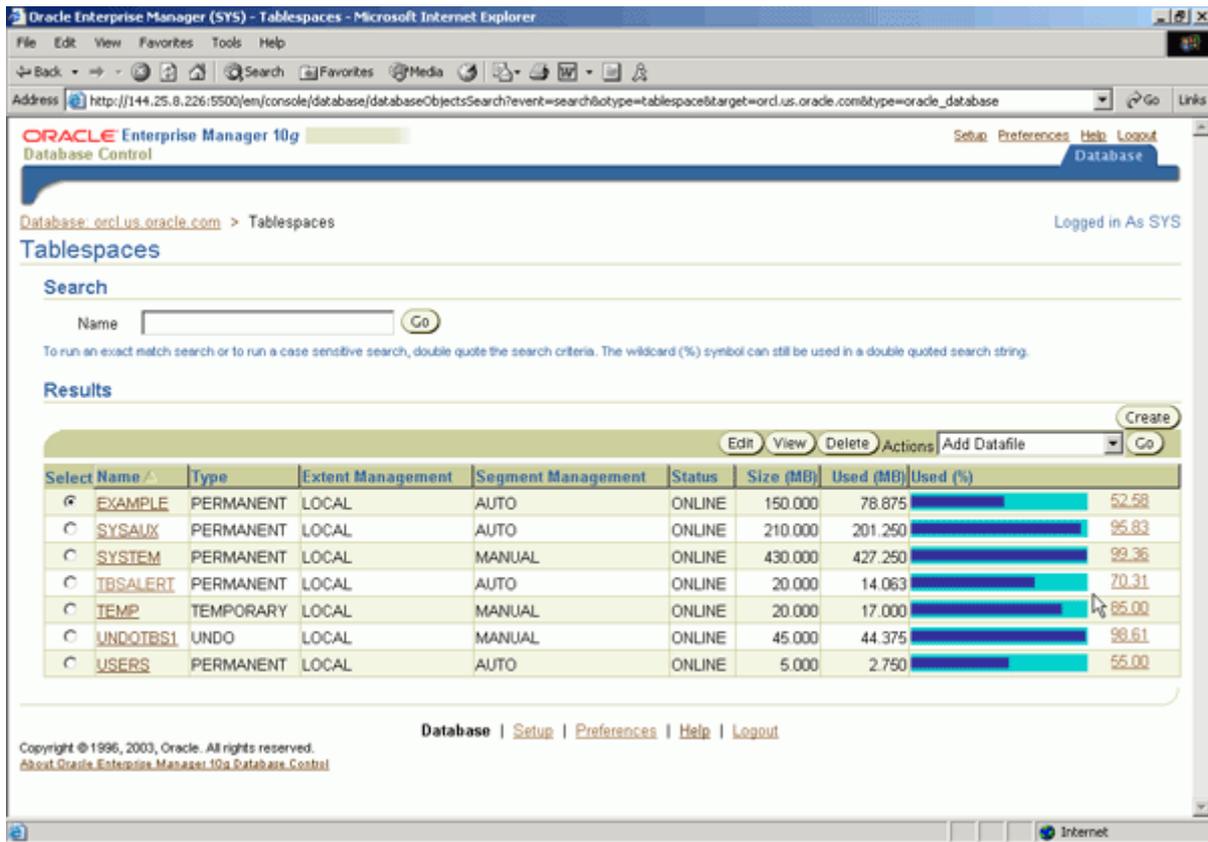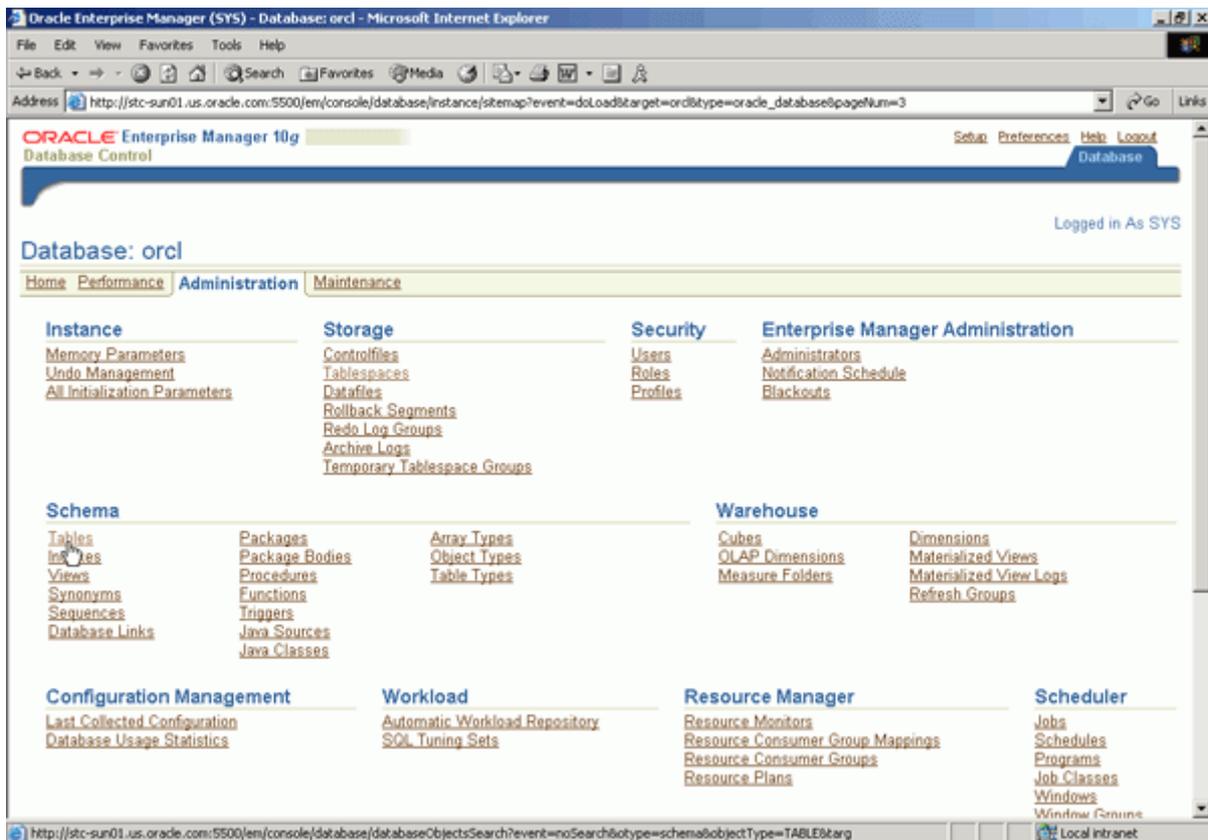
9. 转至 Enterprise Manager 窗口。 刷新浏览器（对于 Linux Mozilla，从菜单栏中选择 View ，然后选择 Reload ）。 注意， TBSALERT 表空间的空间使用率现在已经超过了临界阈值水平 (68%)。

10. 在您等待空间使用率警报显示在 Enterprise Manager 主页上时，请仔细查看表段统计数据。 单击 Database 导览栏，然后单击 Tables 链接。
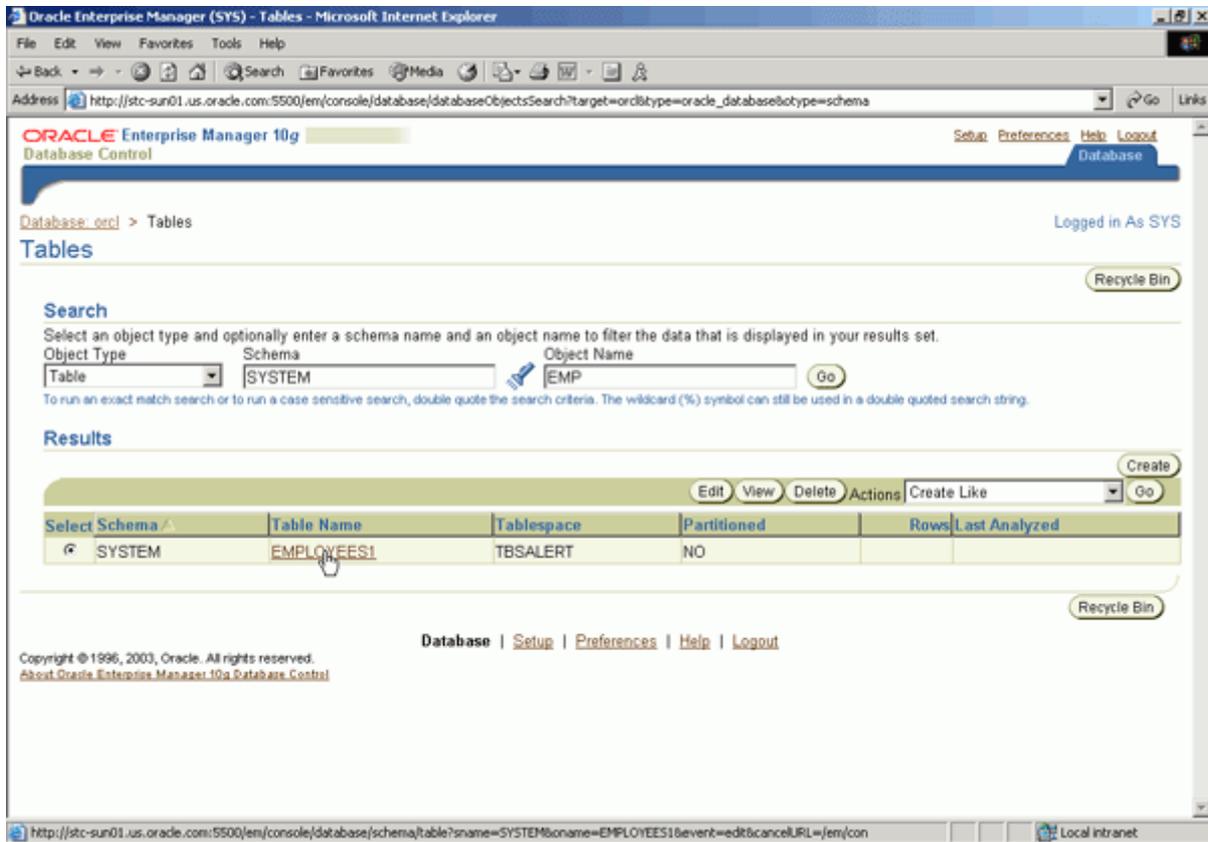
11. 要查找 SYSTEM.EMPLOYEES1 表，请在 Schema 域中输入 system ，在 Object Name 域中输入 emp 。
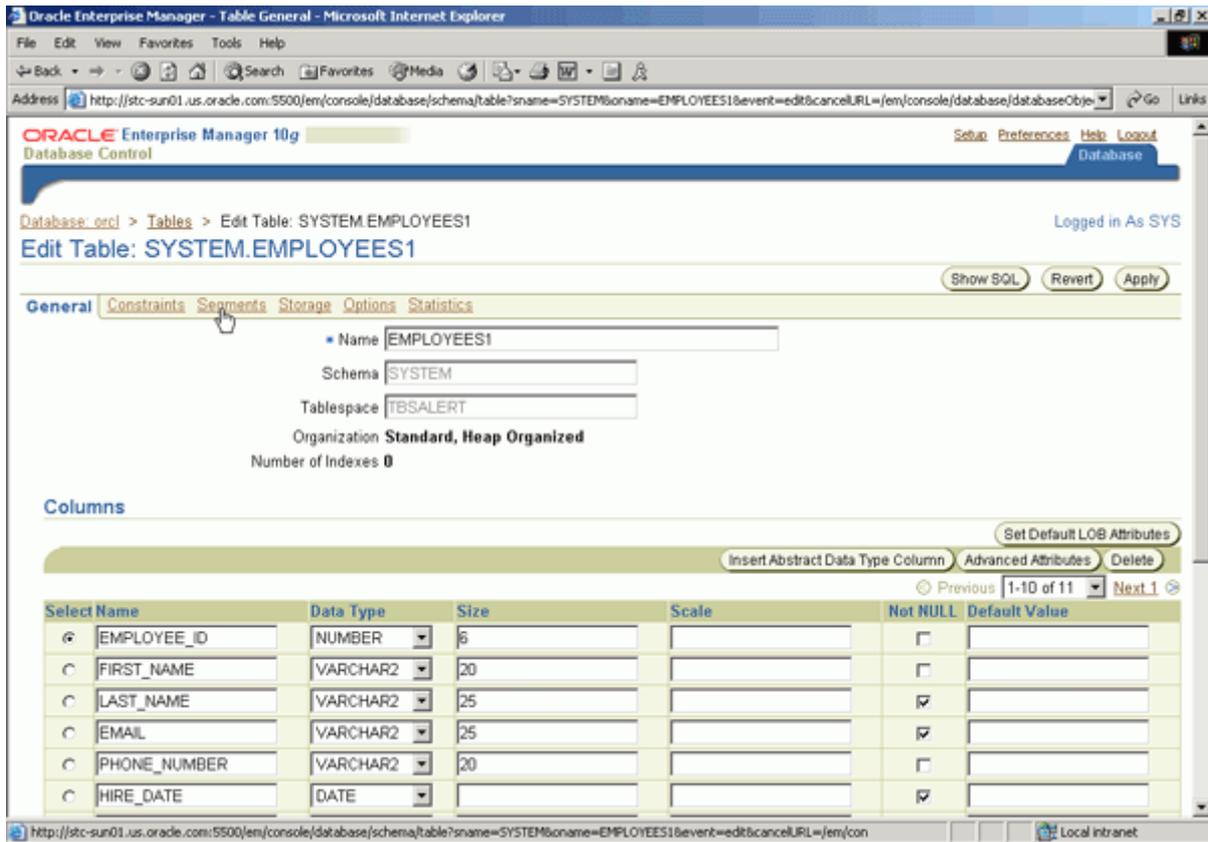
单击 Go 。

12. 单击 EMPLOYEES1 链接。
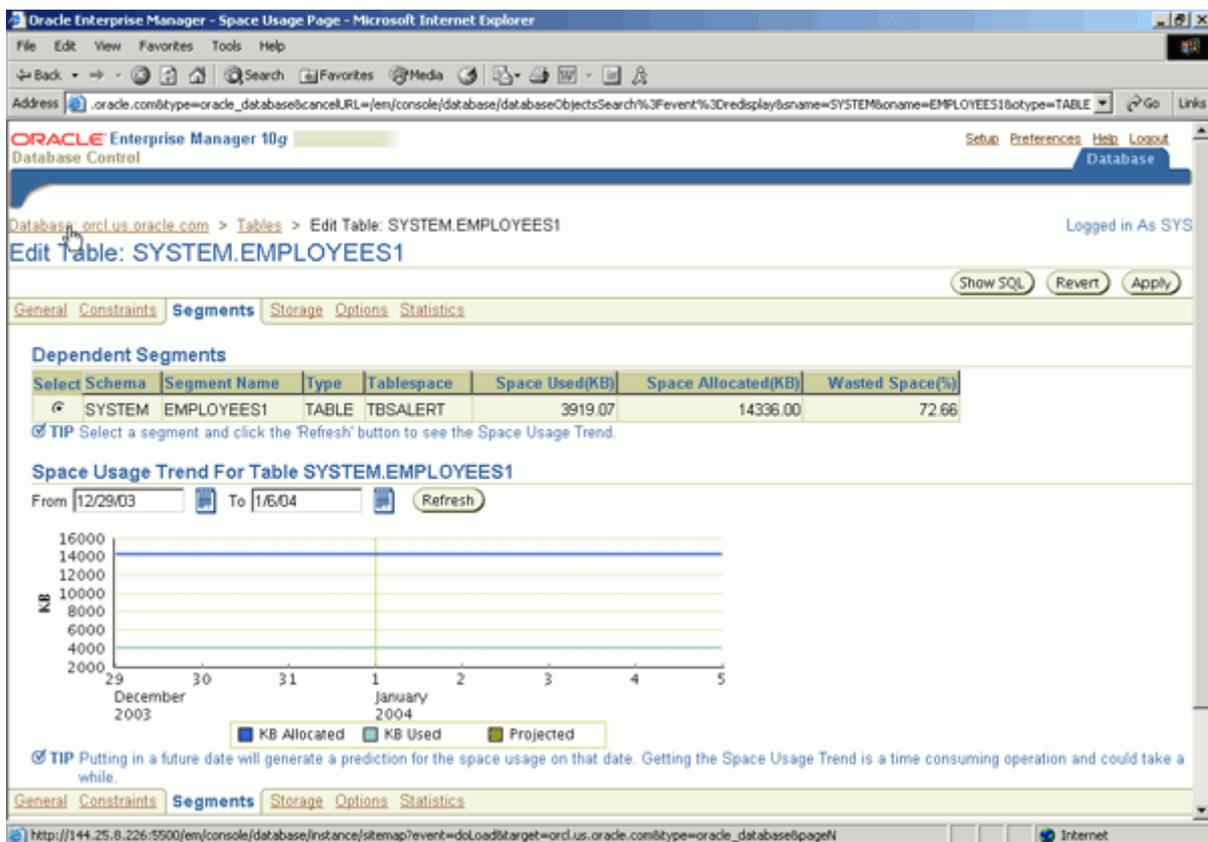
13. 单击 Segments 。
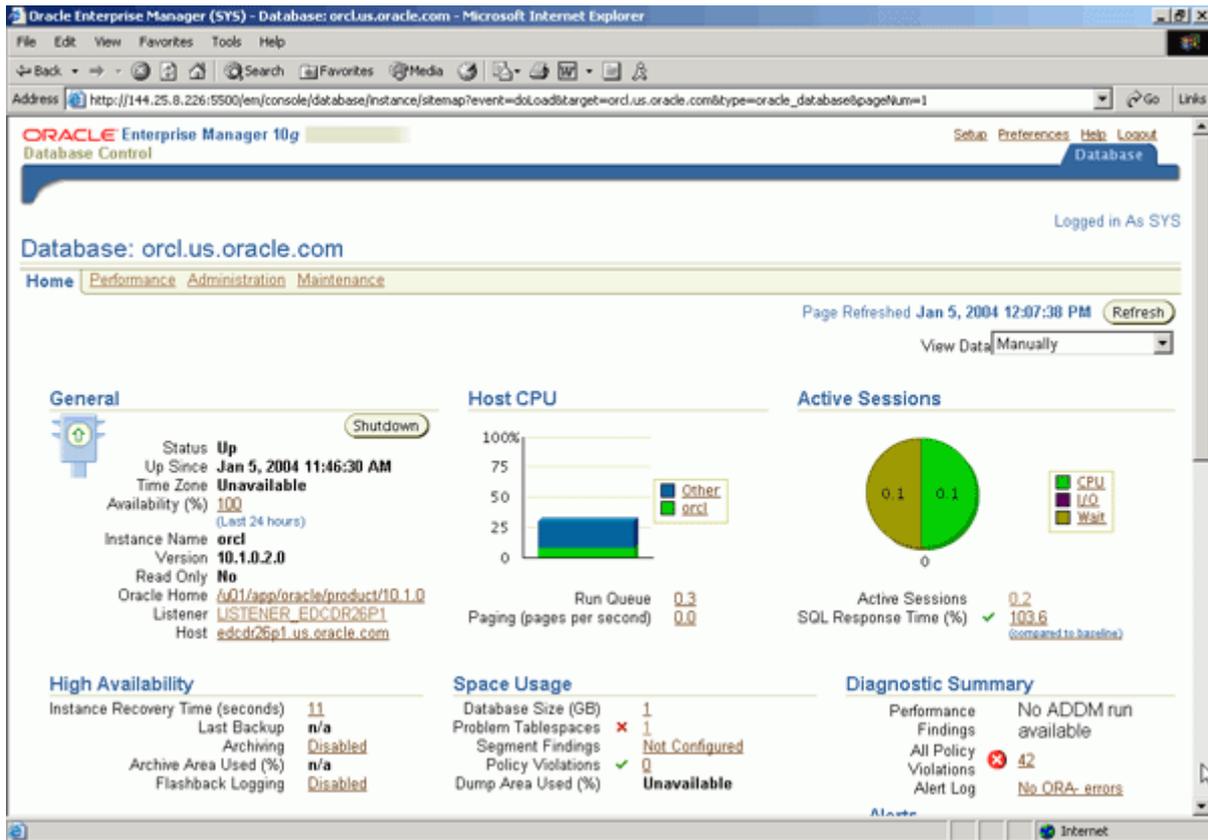
14. 注意 EMPLOYEES1 表中浪费空间的百分率。 您可能能够通过回收表中未使用的空间来解决表空间空间使用率警报。

在同一页面上，您可以为 Space Usage Trend 指定一个日期范围，然后单击 Refresh 按钮，以规划 EMPLOYEES1 表未来的空间使用率。 因为在 EMPLOYEES1 表上没有足够的活动历史纪录，所以您在空间使用率分析图中将看不到非常有意义的数据。 单击 Database 导览栏，然后单击 Home 页面标签。
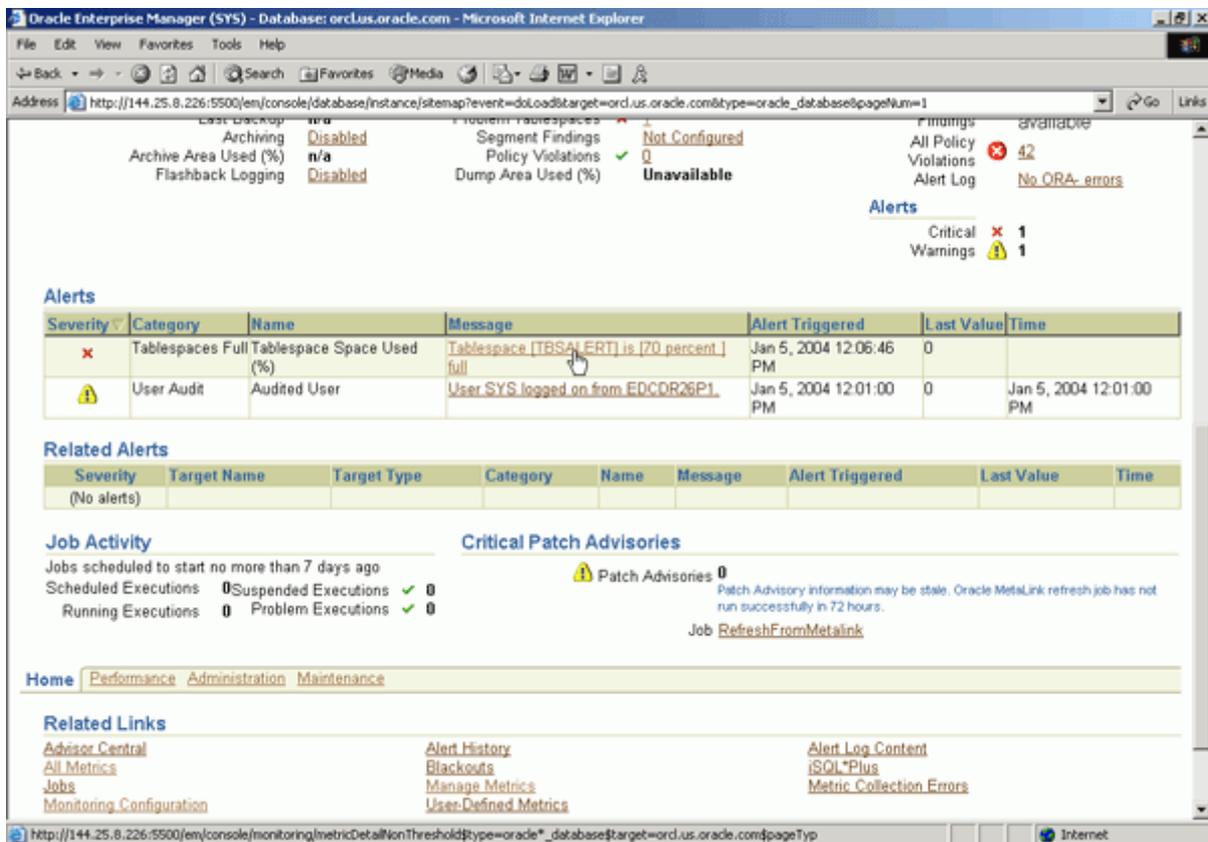
15. 单击 Refresh 按钮几次，直到您看到一个红色的 x 和数字 1 紧邻着 Problem Tablespaces 出现。 向下滚

动至 Alerts 表。
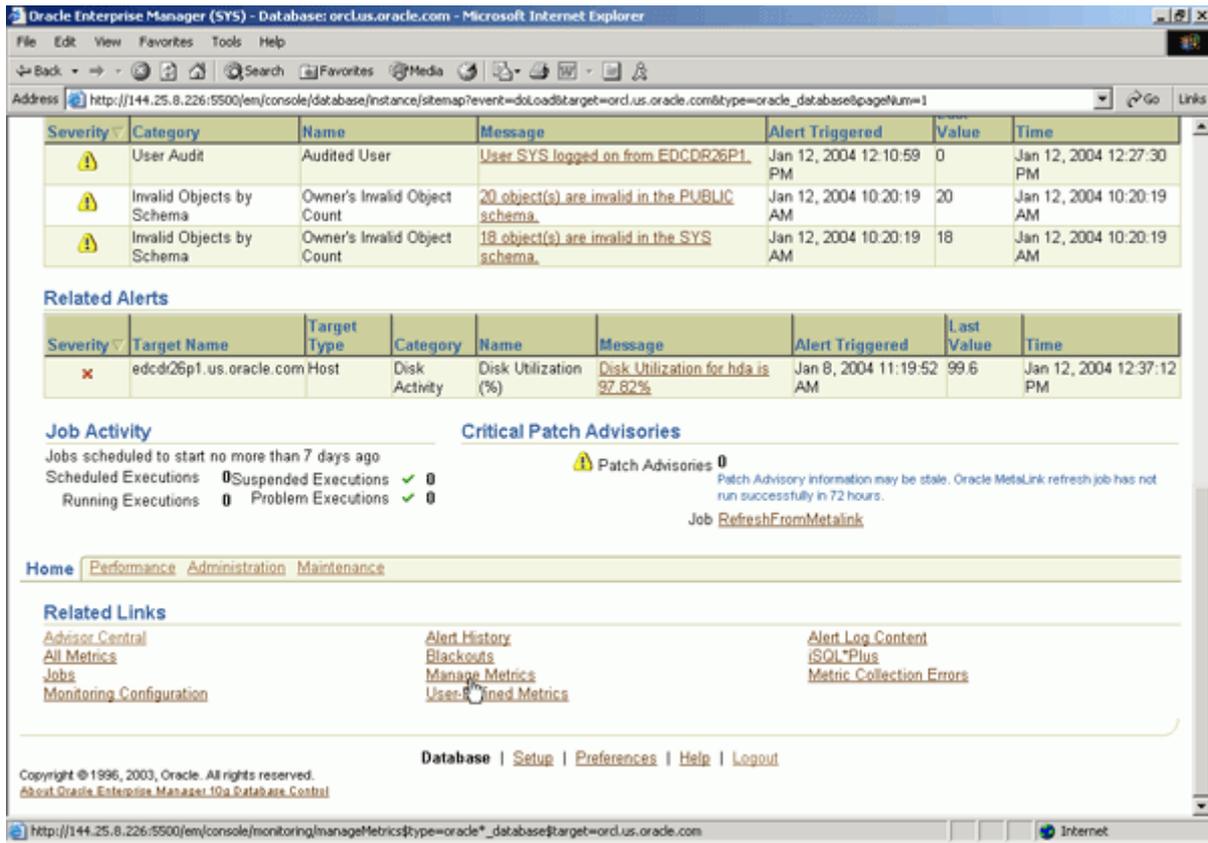
16. 您将看到一个 Tablespaces Full 警报。 单击 Tablespace [TBSALERT] is [70 percent] full 链接。
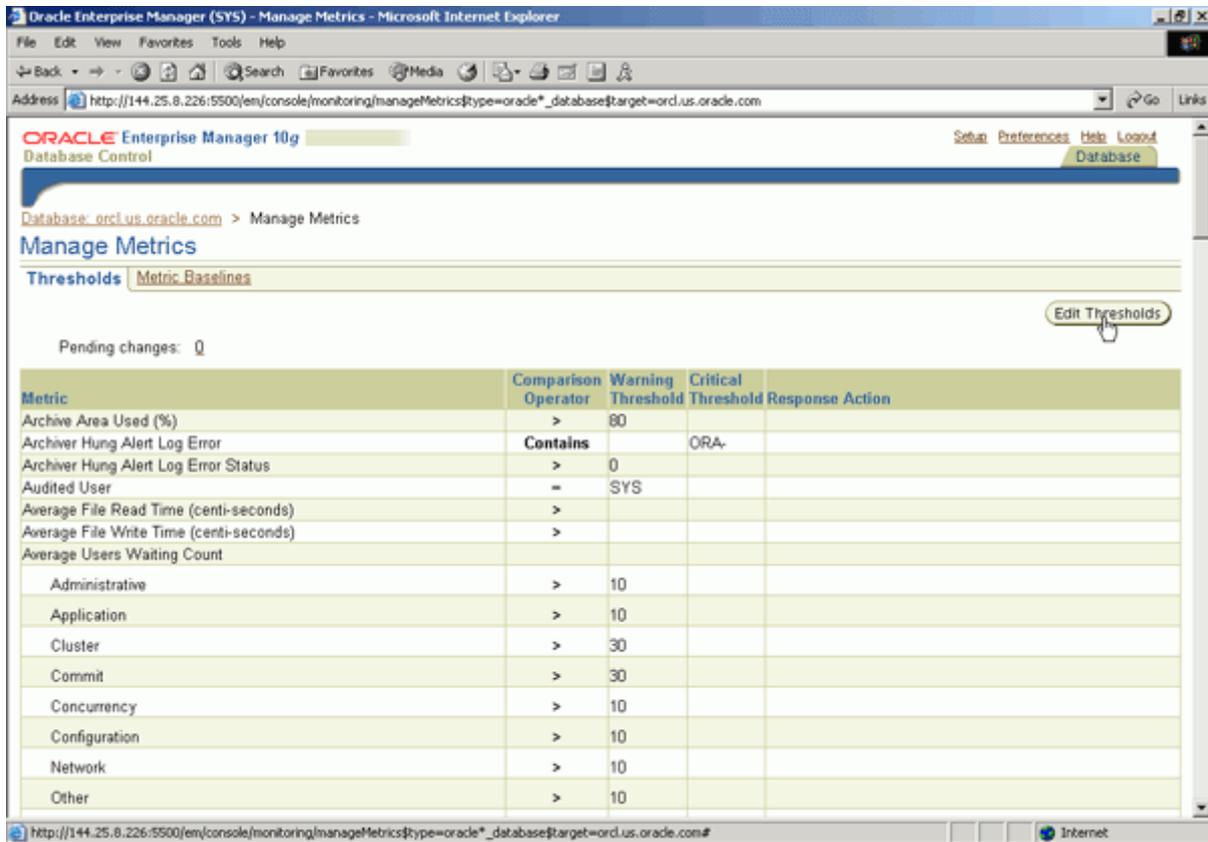
设置量度阈值

　　Oracle 提供了一组预先定义的量度，其中一些量度一开始就拥有为它们定义的阈值。 您将预先为 TBSALERT 表空间的表空间使用率定义一个量度。 要查看所有的量度，请执行以下操作:
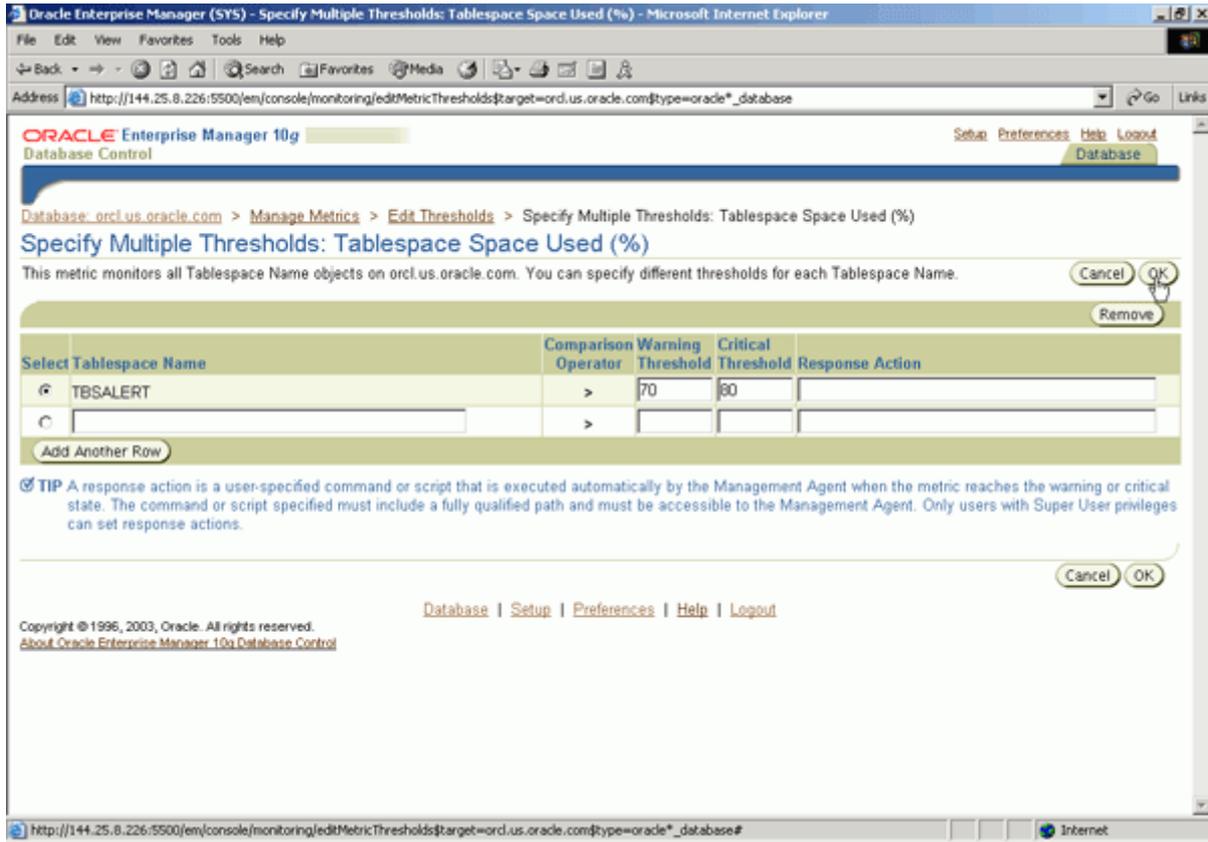
1. 单击 Related Links 区域中的 Manage Metrics 。
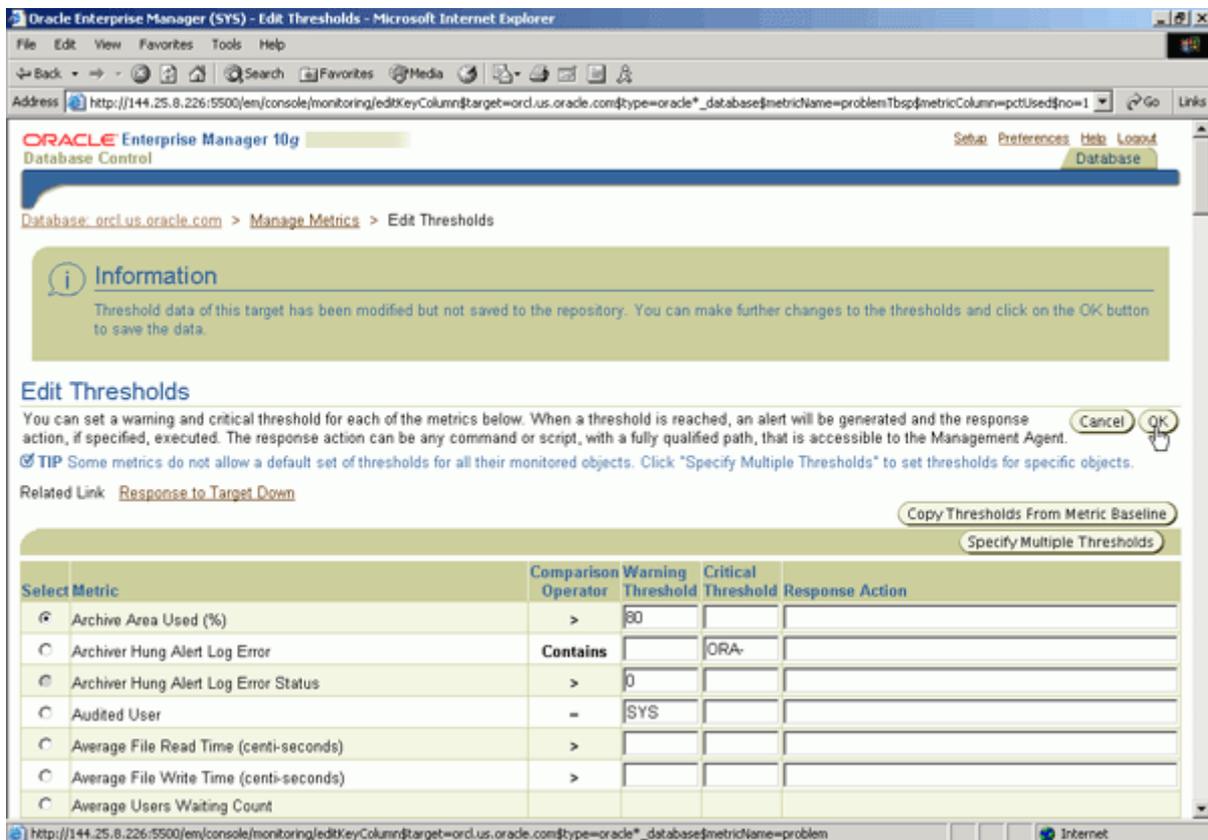
2. 单击 Edit Thresholds 。
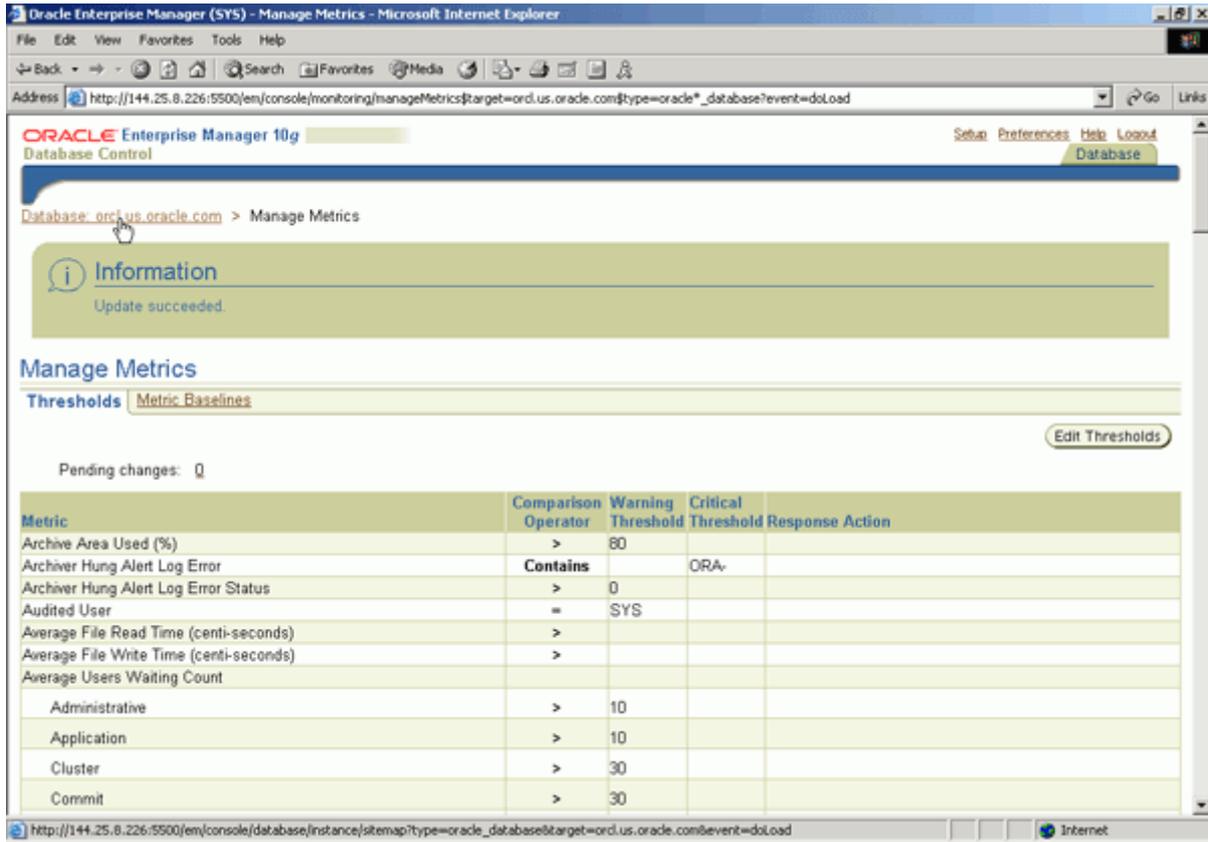
3. 向下滚动至 Tablespace Space Used (%) ，然后选择这个量度。 向上滚回到窗口顶部。

4. 单击 Specify Multiple Thresholds 。

5. 对于 TBSALERT 表空间,将警告阈值修改为 70 ，将临界阈值修改为 80。 然后单击 OK 。

6. 修改完成。 单击 OK ，将数据保存到数据库中。
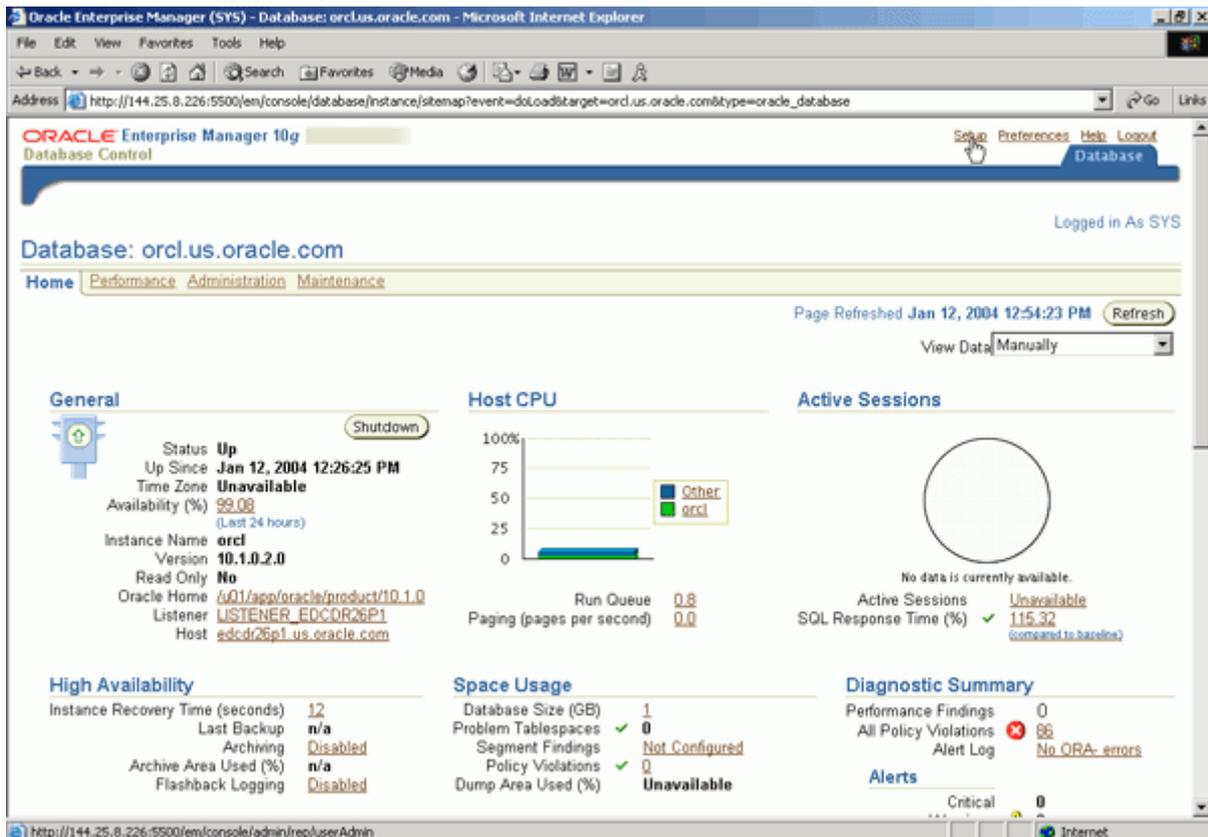
7. 更新成功。 单击路径式导航栏中的 Database 导航项。
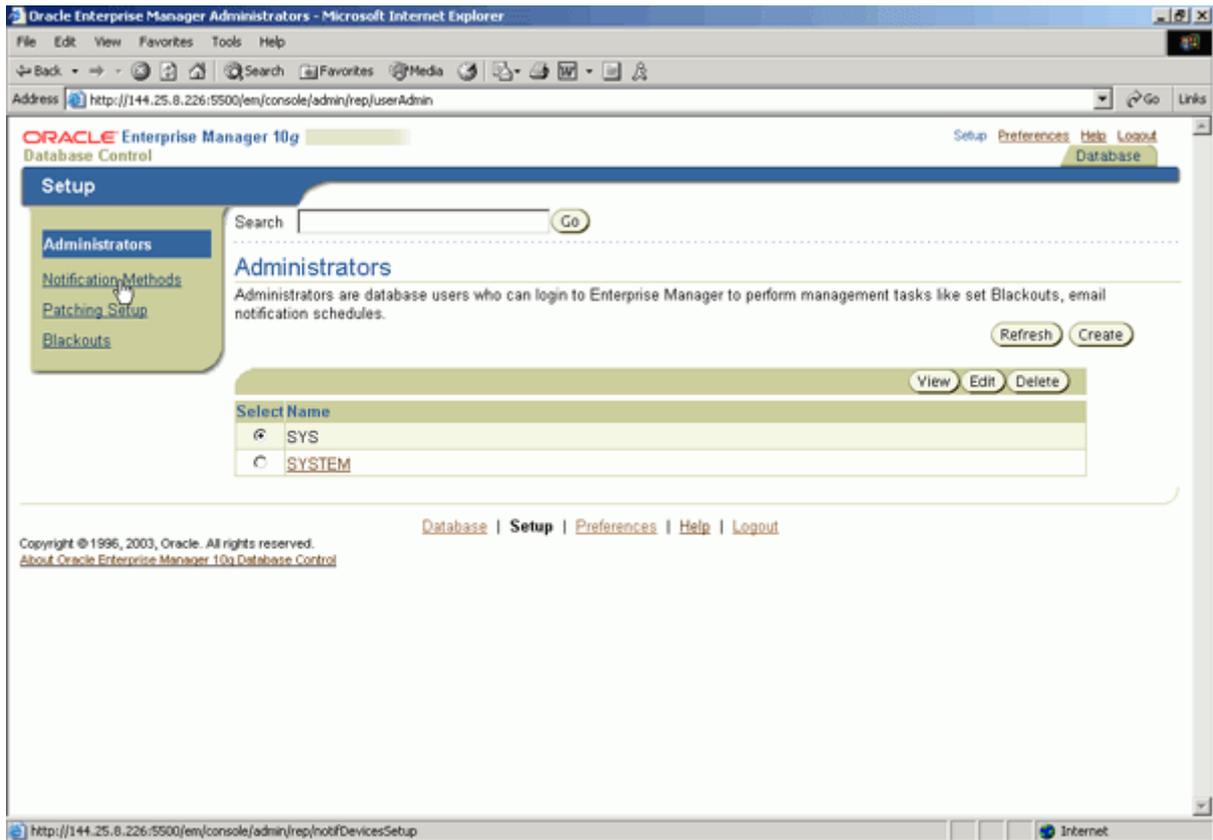
设置通告

当出现需要您干预的事件时，您可以选择性地提供通告。 默认地，临界状态的警报（如 Database Down、Generic Alert Log Error Stats 和 Tablespace Used）都设置为提供通告。 执行以下操作：
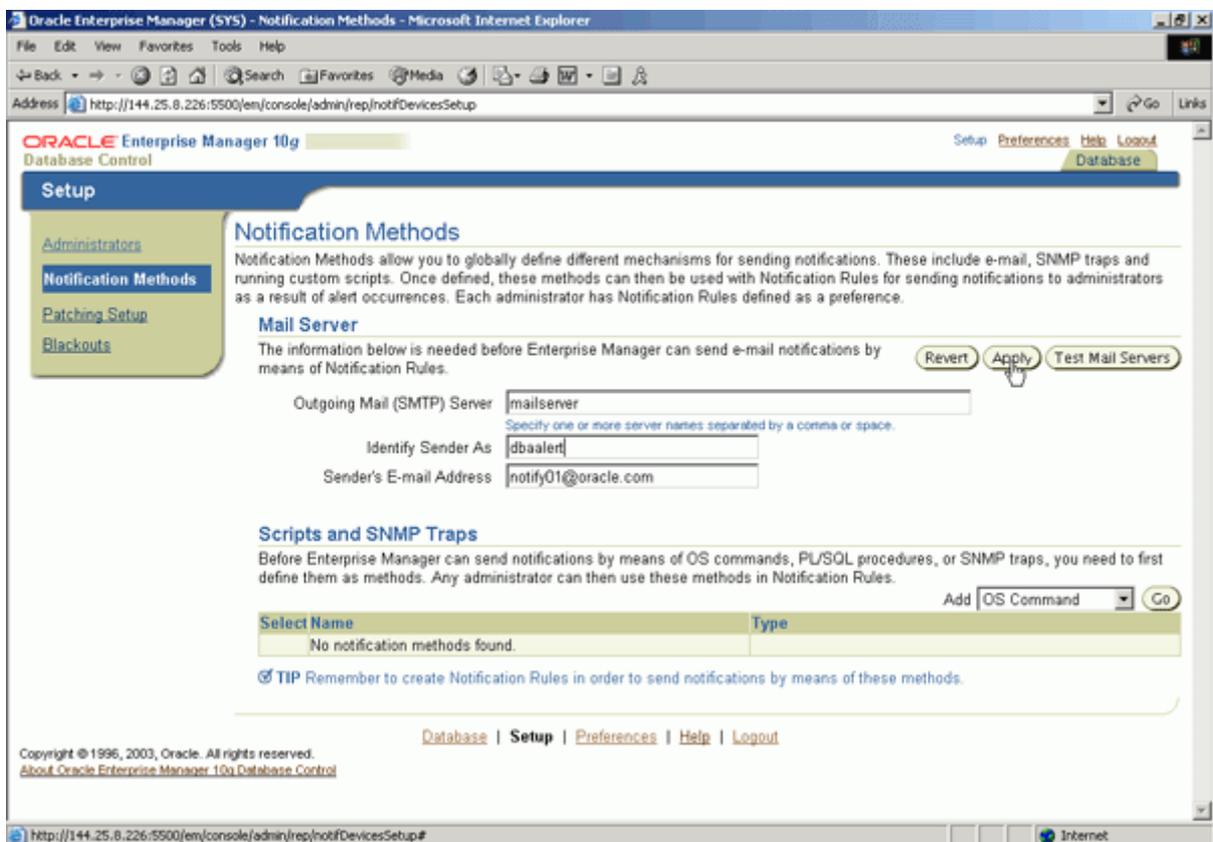
1. 单击 Database 主页顶部的 Setup 。
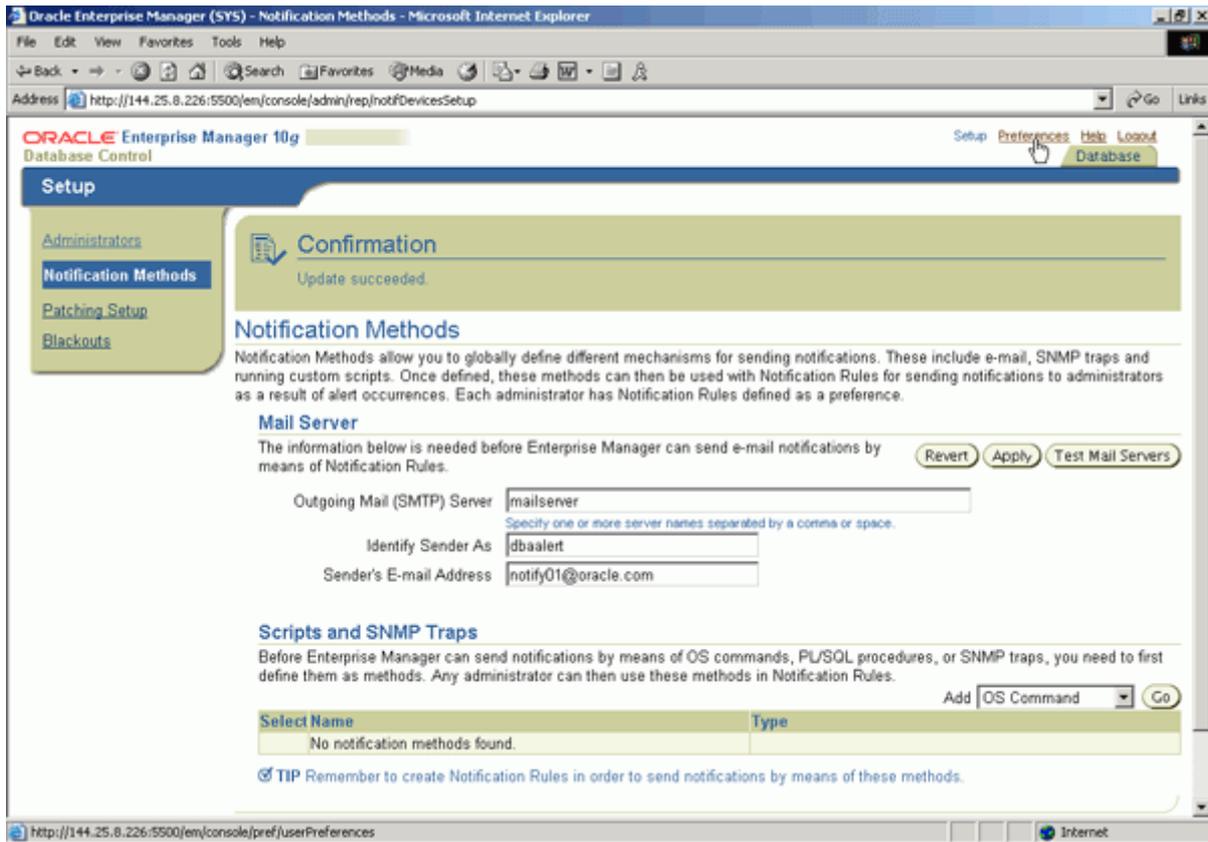
2. 单击 Notification Methods 。
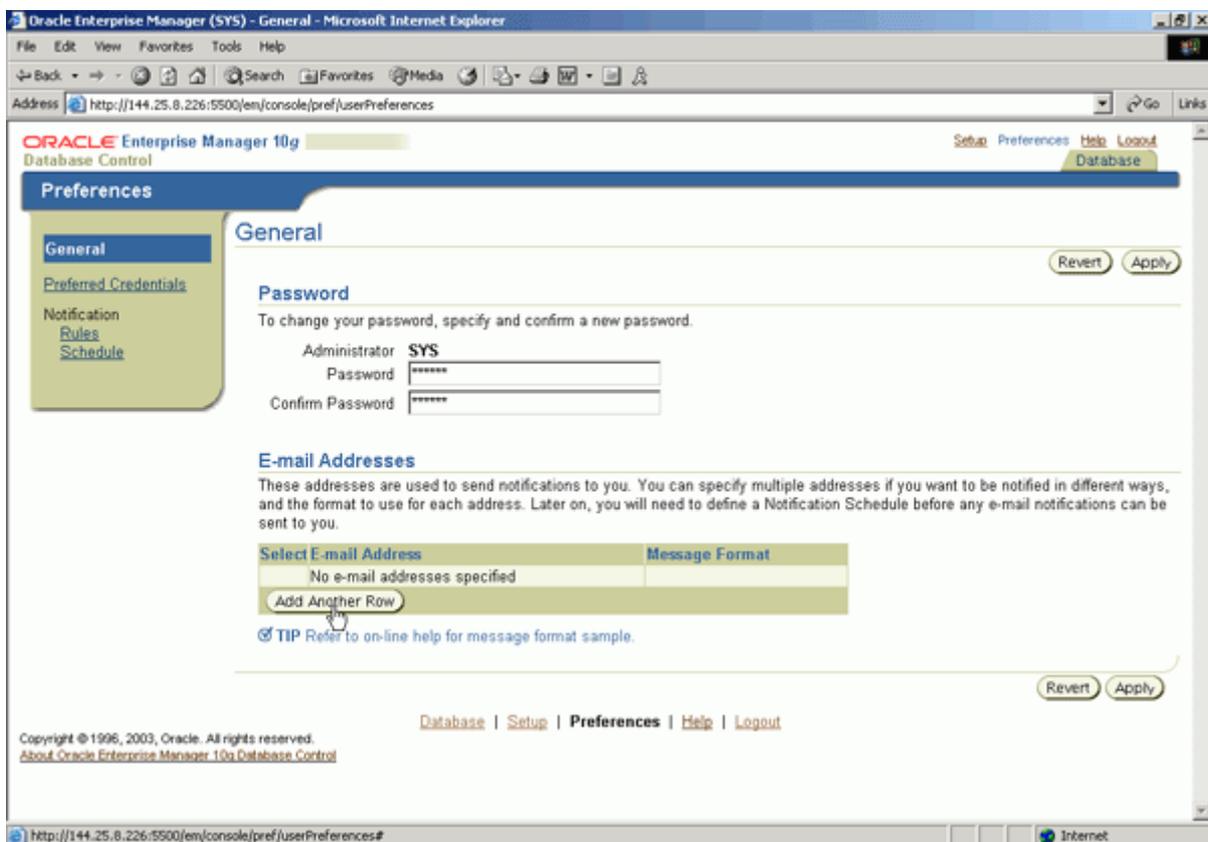
3. 在 Outgoing Mailserver 域中输入 <您的邮件服务器> ，在 Identify Sender As 域中输入 dbaalert ，在 Sender's Email 域中输入 notify01@oracle.com ，然后单击 Apply。
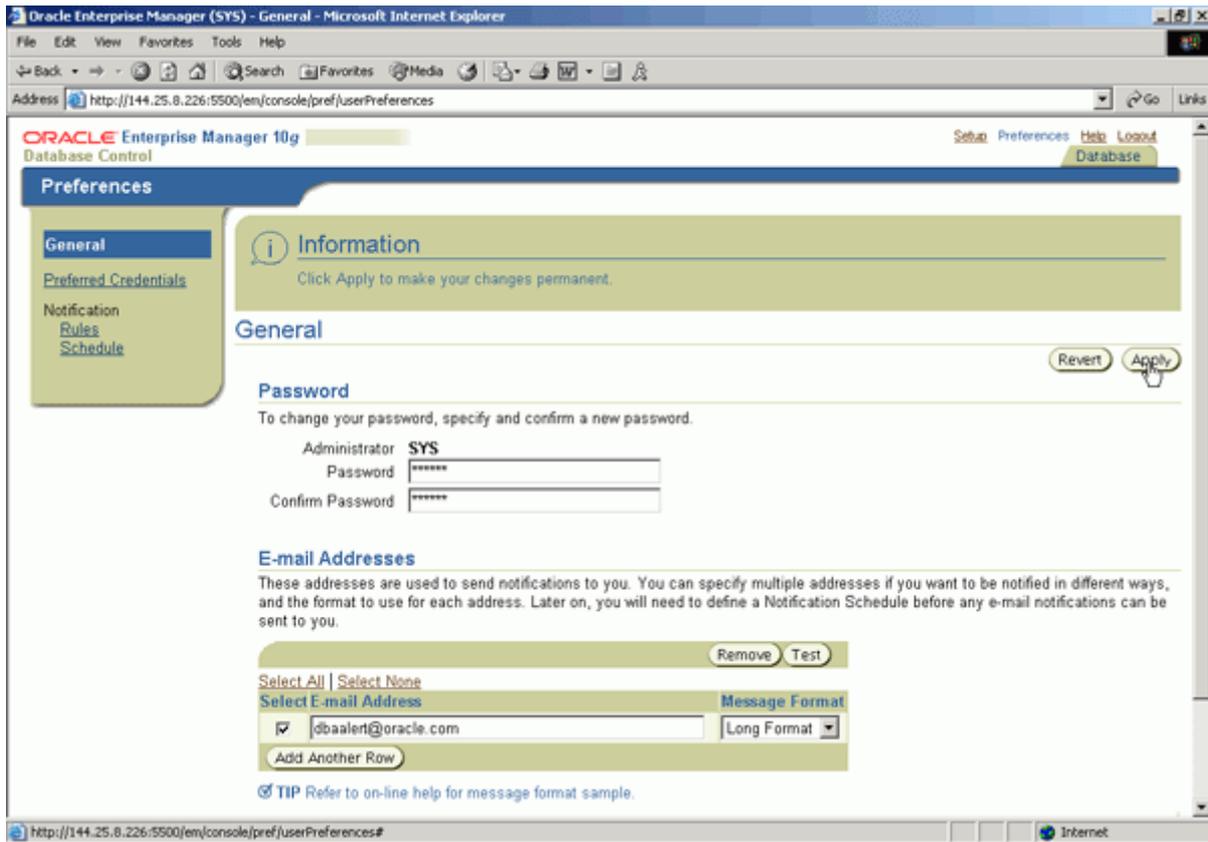
4. 更新成功。 单击页面顶部的 Preferences 。

5. 单击 General 选项中 E-mail Addresses 的 Add Another Row 。

6. 输入 notify01@oracle.com 作为电子邮件地址，单击 Apply 。 然后单击 Database 。

诊断和解决性能问题

在出现数据库性能问题时，需要您进行诊断和纠正。 有时用户抱怨性能缓慢会使您注意到问题。 其余的时候，您可能注意到主页上的 Host CPU 图中的性能尖峰。

在所有情况下，自动数据库诊断监控器 (ADDM) 都将标记这些问题，它默认每 60 分钟执行一次自上而下的系统分析，并在 Oracle Enterprise Manager 主页上报告它发现的问题。 ADDM 每 60 分钟自动运行一次，和自动工作负载信息库 (AWR) 采集的快照同步发生。 它的输出包含了它确认的每个问题的说明和建议操作。
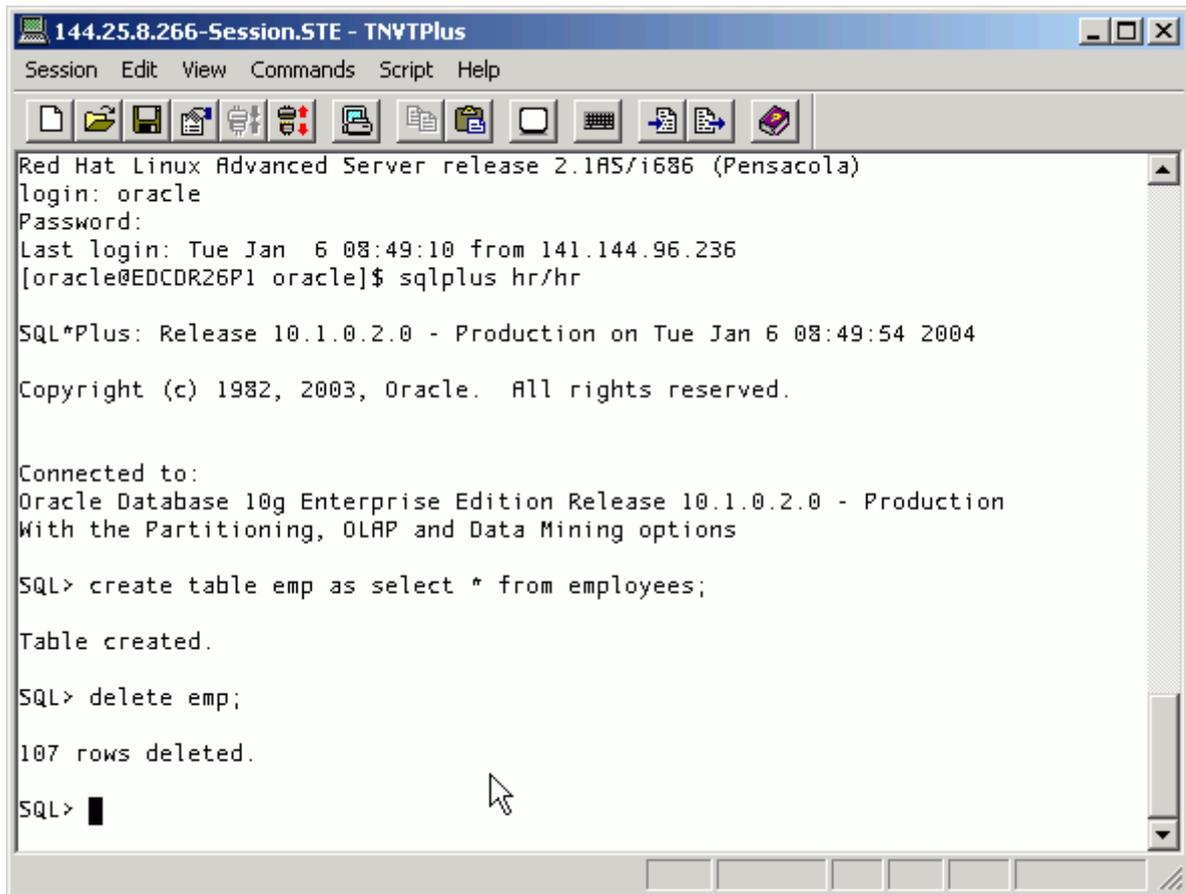
- 生成性能问题查找结果
- 利用 ADDM 解决发现的性能问题

生成性能问题查找结果

为了演示 ADDM 如何工作，您要生成一个性能问题查找结果。 在本文中，您将创建一个等待行锁的会话。要执行某些如更新和删除的操作，会话必须锁住该行。 执行以下步骤，生成性能问题查找结果：

1. 打开终端窗口， 执行以下命令：

sqlplus hr/hr create table emp as select * from employees; delete emp;
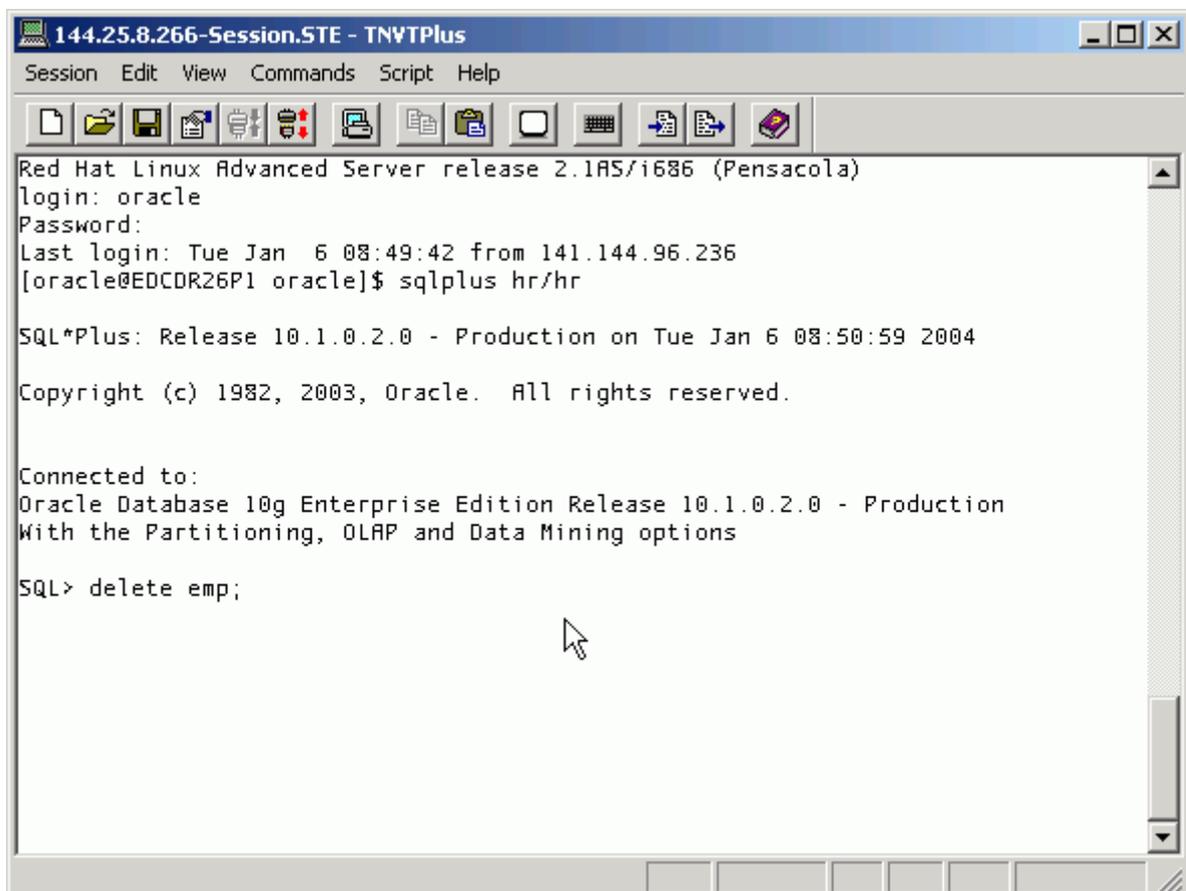
2. 再打开一个终端窗口，并执行以下命令，创建一个行锁冲突：

sqlplus hr/hr delete emp;
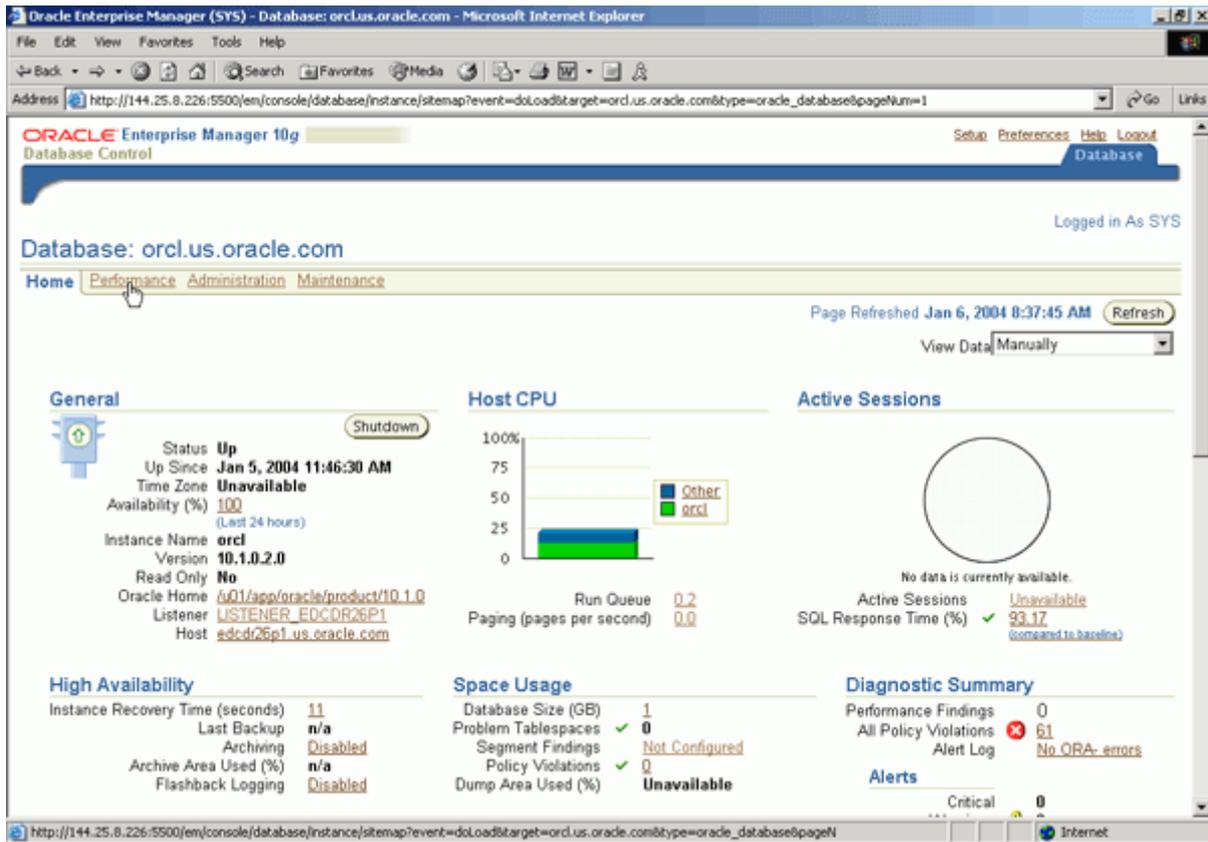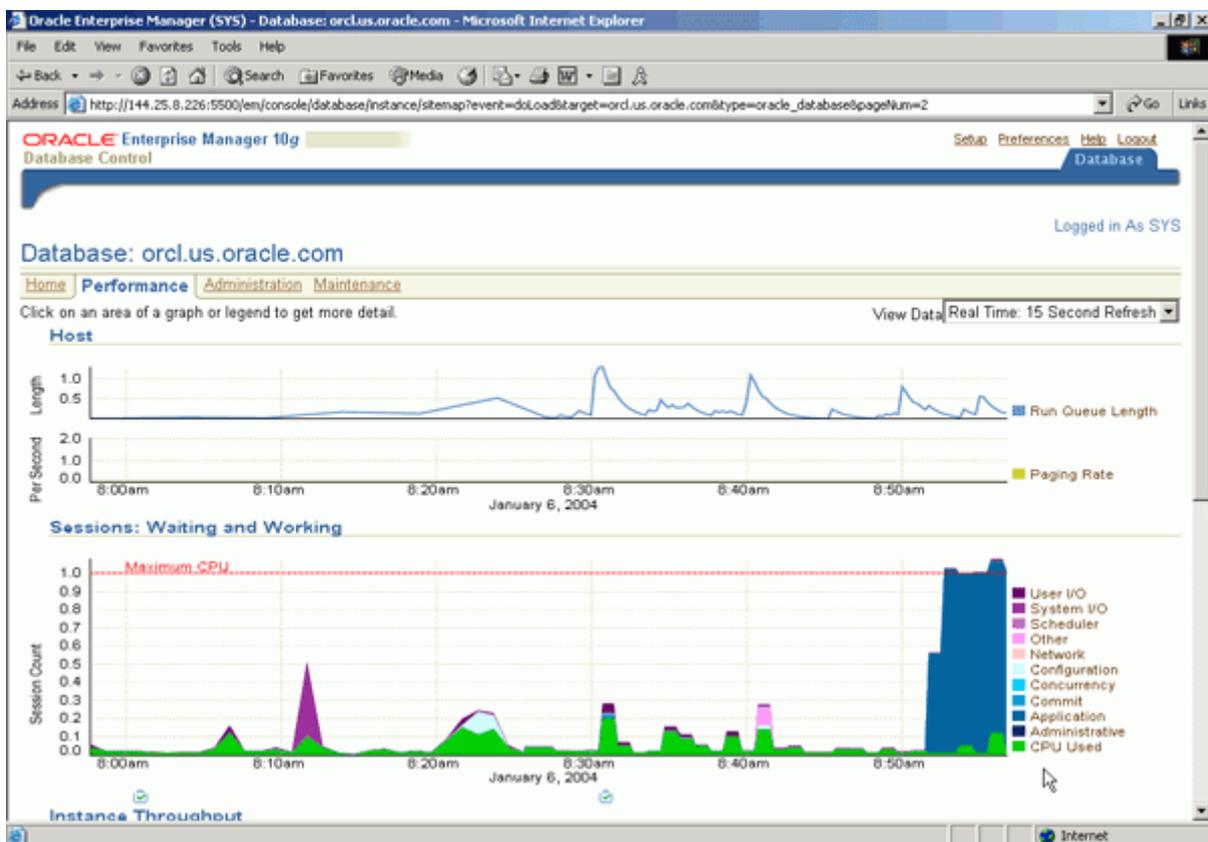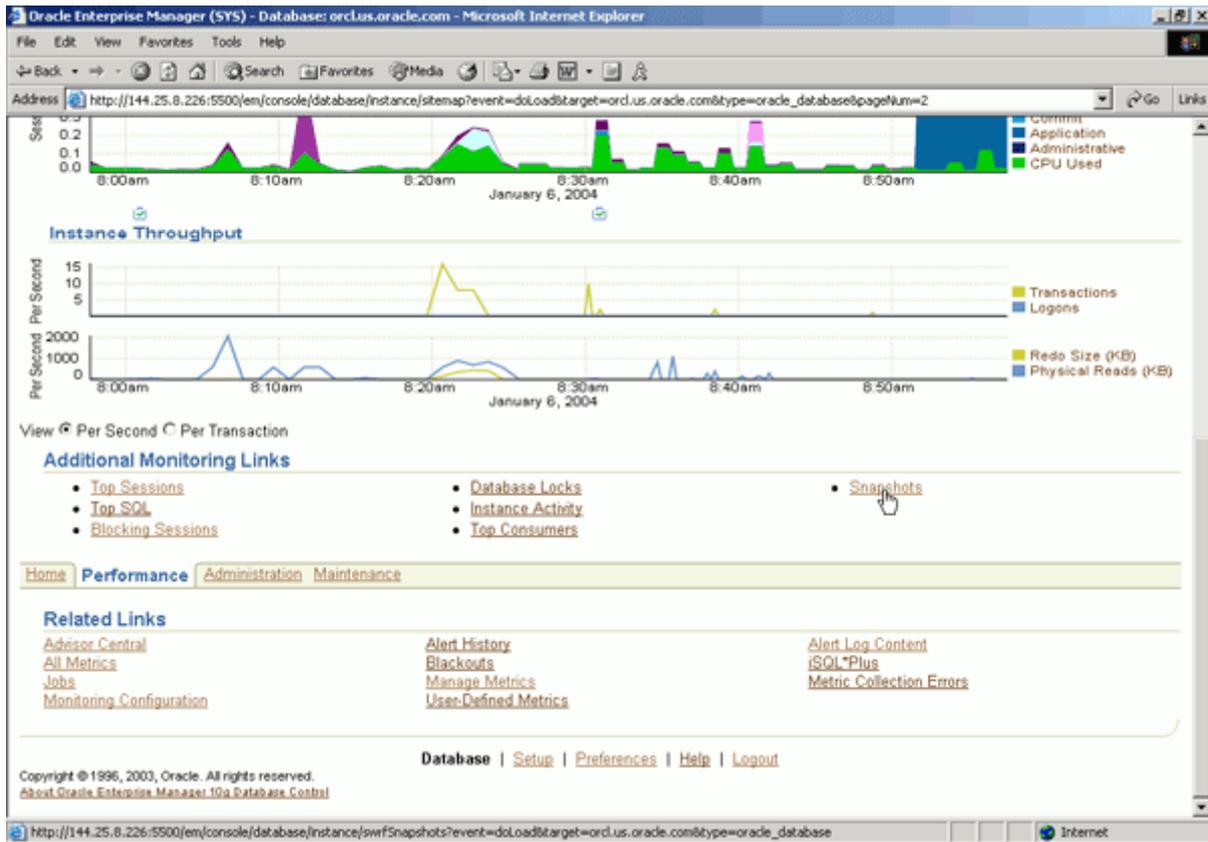
3. 单击 Enterprise Manager 窗口中的 Performance 。
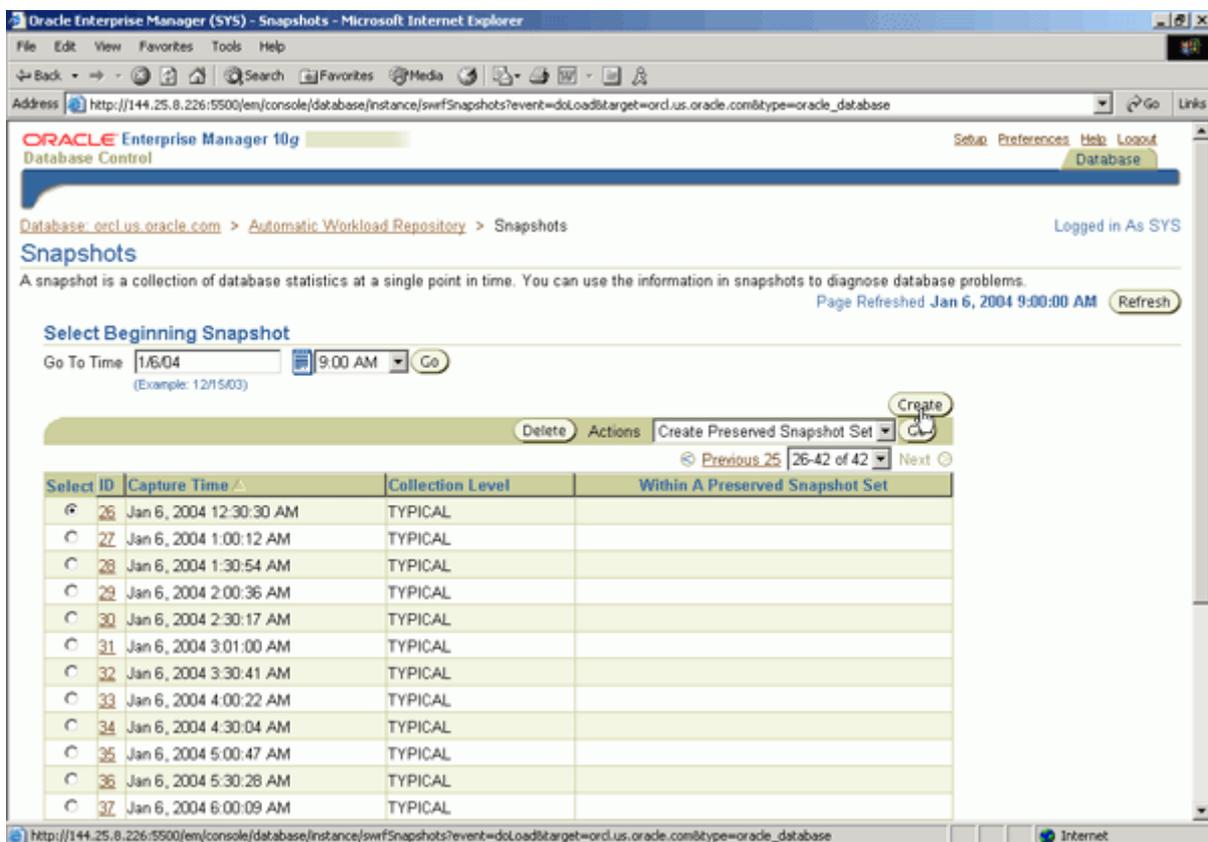
4. 您将看到会话等待值非常高。 等待大约 10 分钟，然后向下滚动至窗口底部。
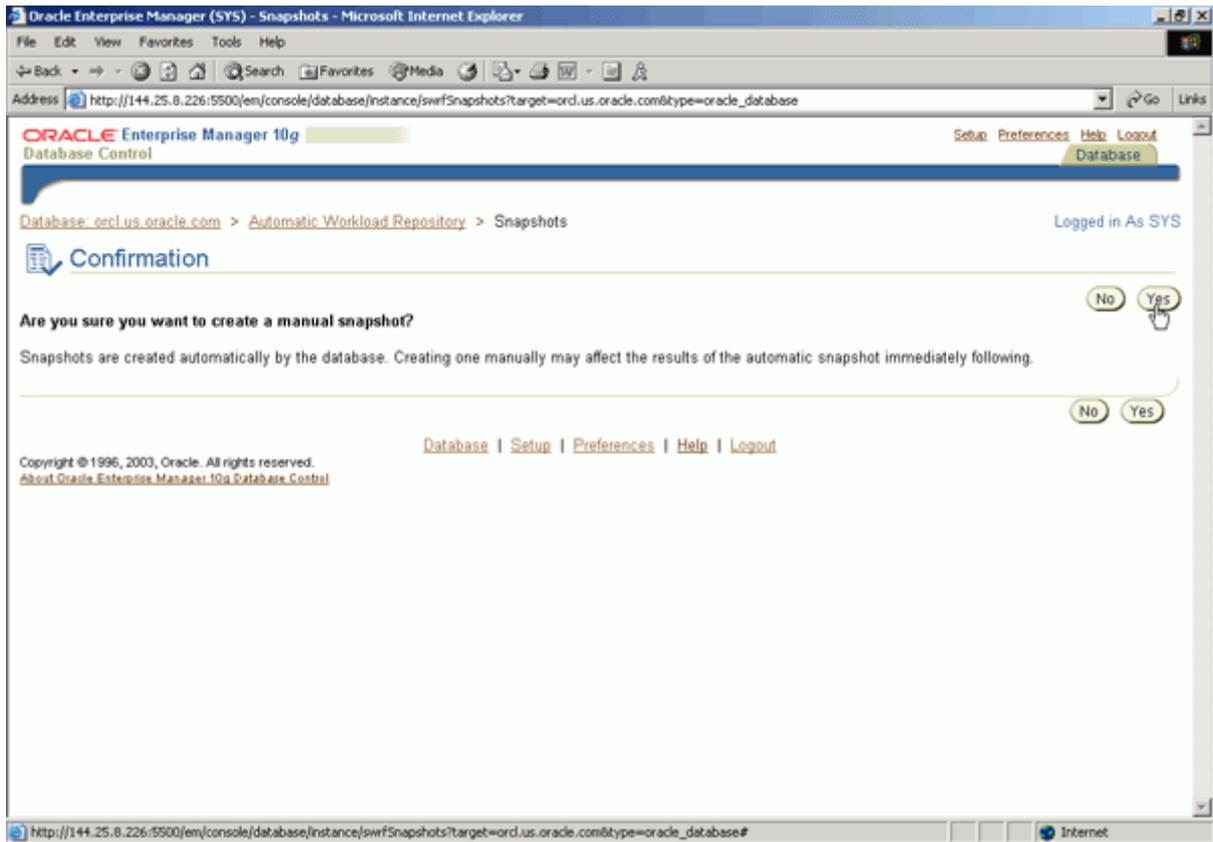
5. 您现在将创建快照来捕获性能问题。 单击 Snapshots 。
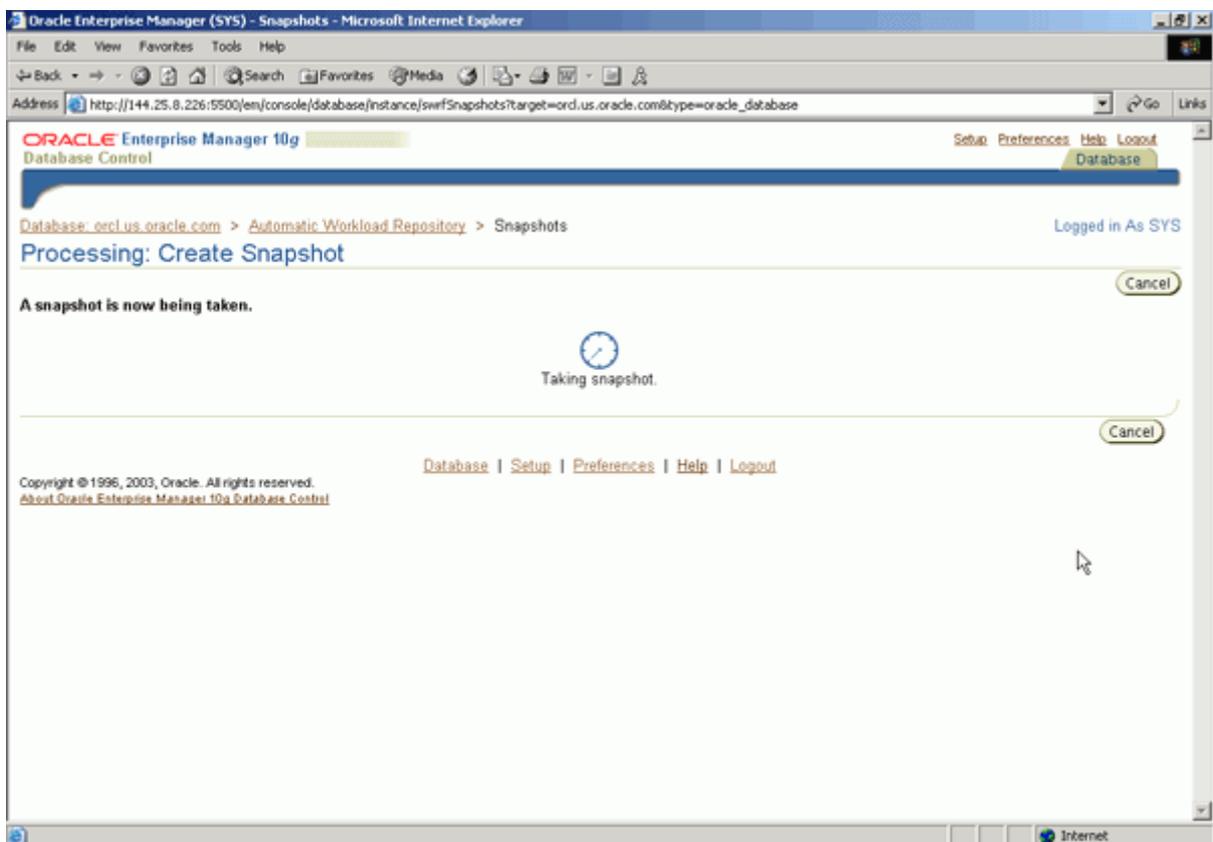
6. 单击 Create ，创建快照。
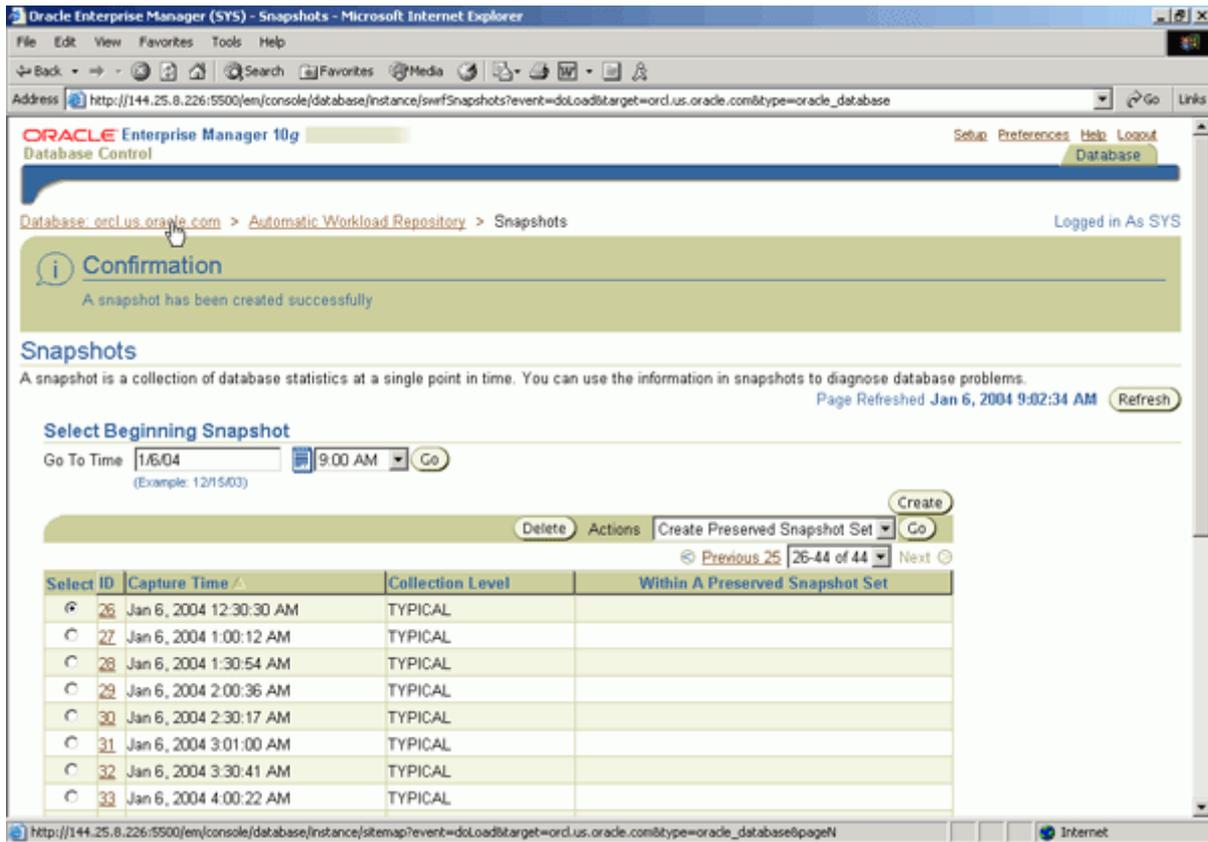
7. 单击 Yes ，创建人工快照。
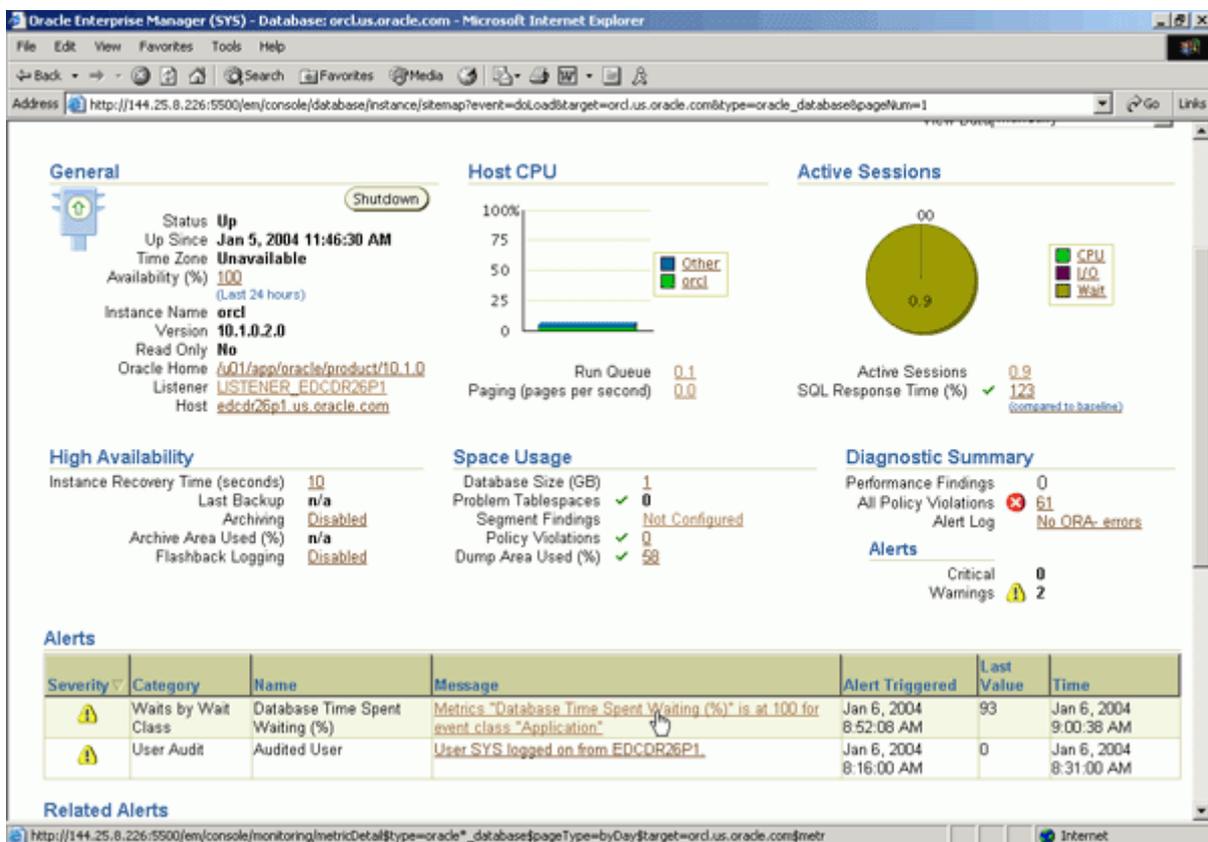
8. 正在创建快照。

9. 一旦快照创建完成，请单击路径式导航栏中的 database 导航项，然后单击 home 。
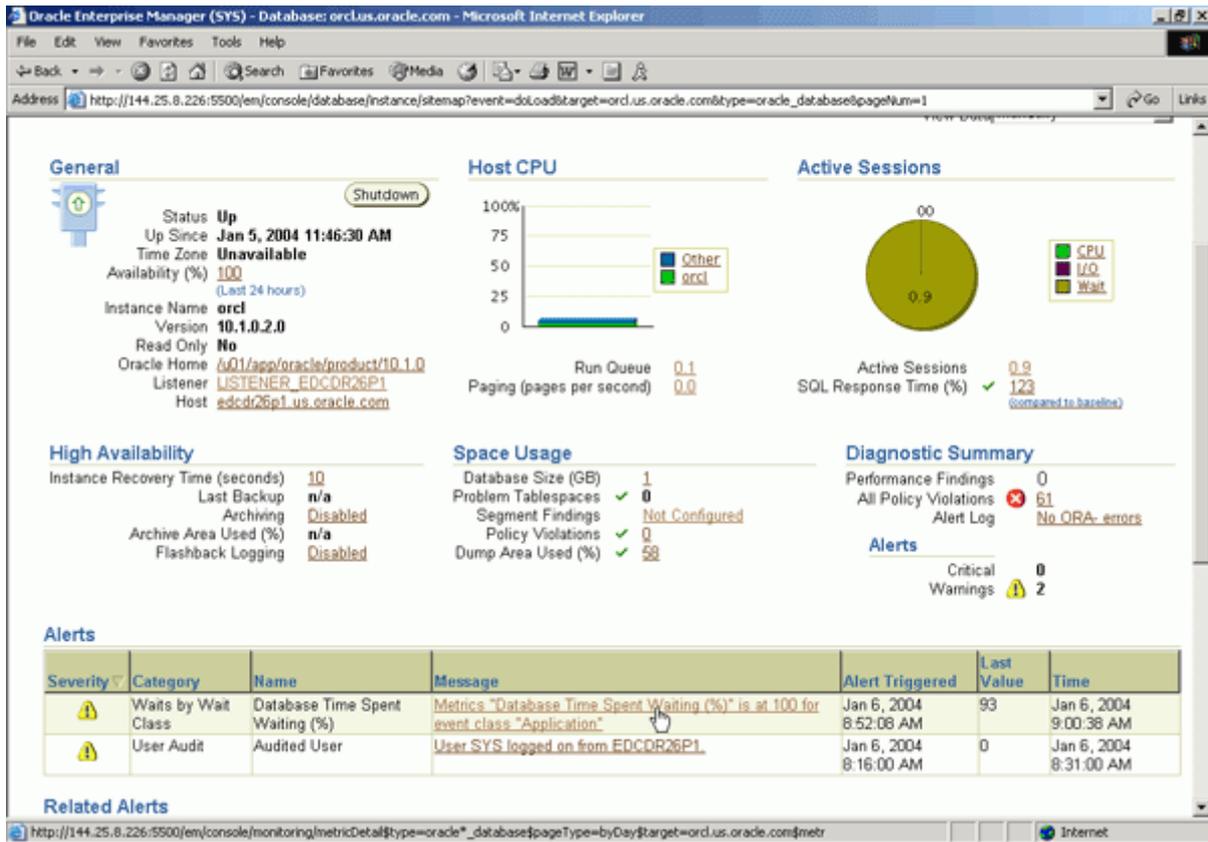
10. 现在通过主页的 Alert 部分的一个警报,检测到了一个性能问题。
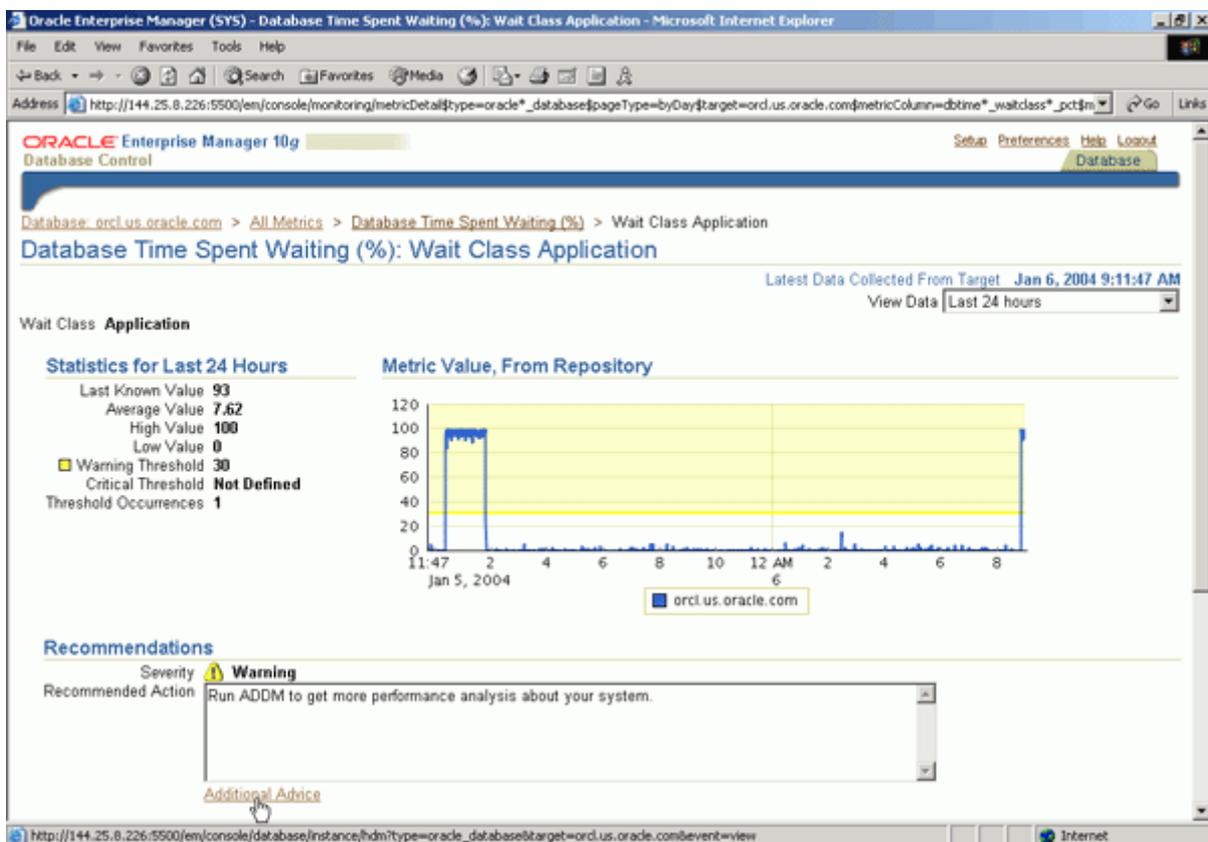
利用 ADDM 解决发现的性能问题

当遇到性能问题时,您可以使用 ADDM 来解决它。 执行以下操作:
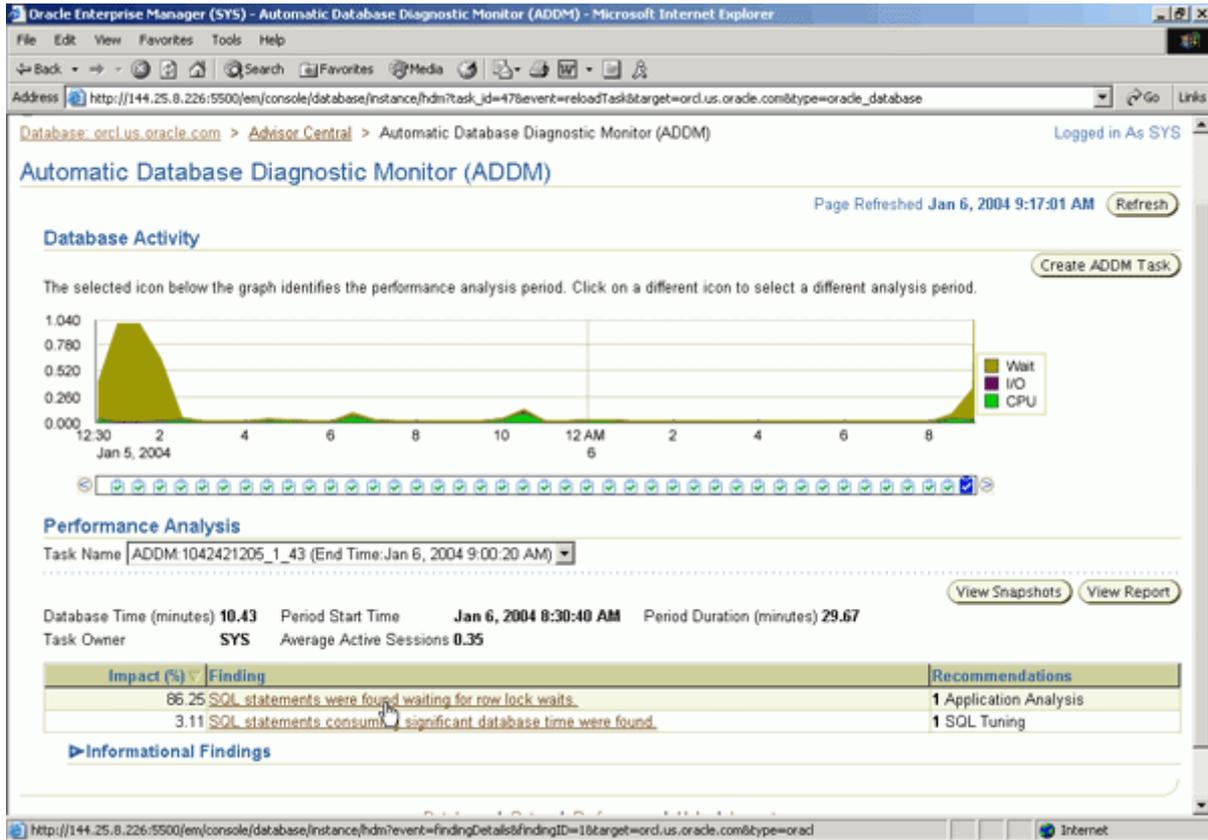
1. 单击警报 Database Time Spent Waiting 。
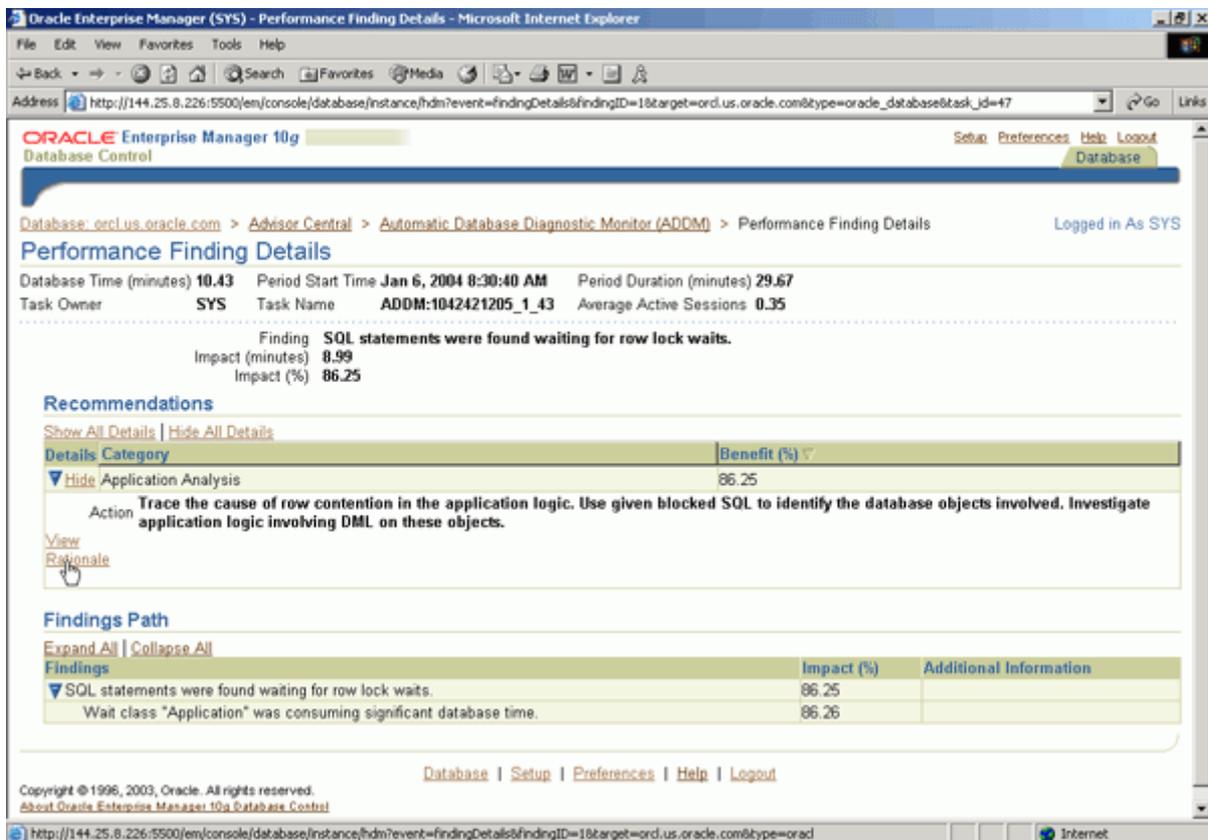
2. 注意建议的操作是运行 ADDM 来获取更多的关于系统的性能分析。 单击 Additional Advice 。
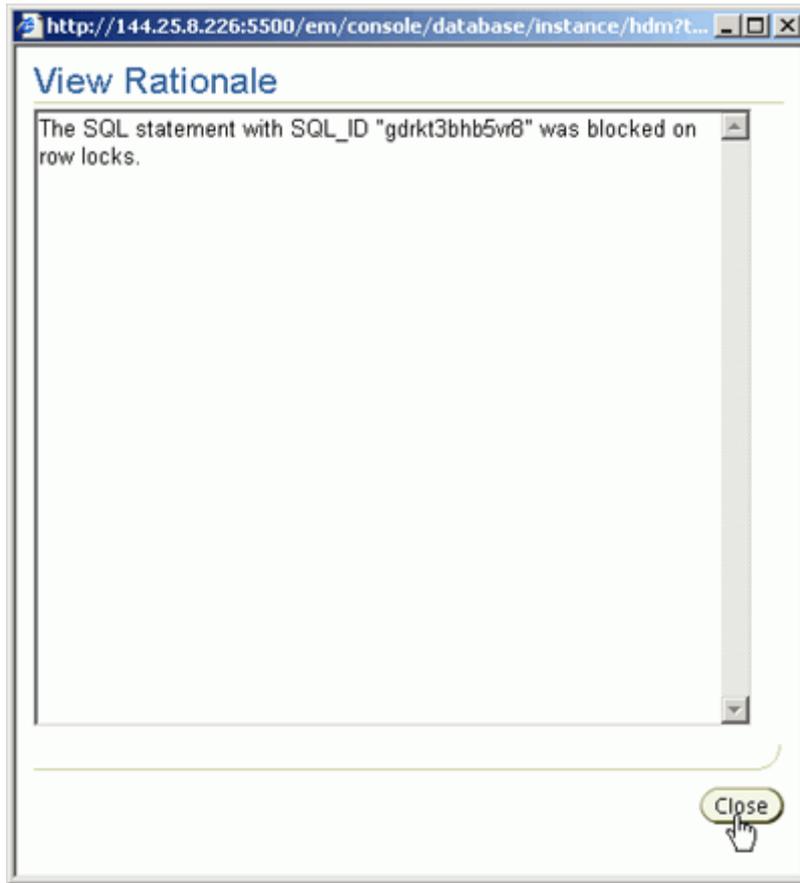
3. 确保从列表中选择了您创建的快照。 注意 SQL statements were found waiting for row lock waits 将影响系统超过 86% 的数据库时间。 单击列表中的这个性能问题查找结果。
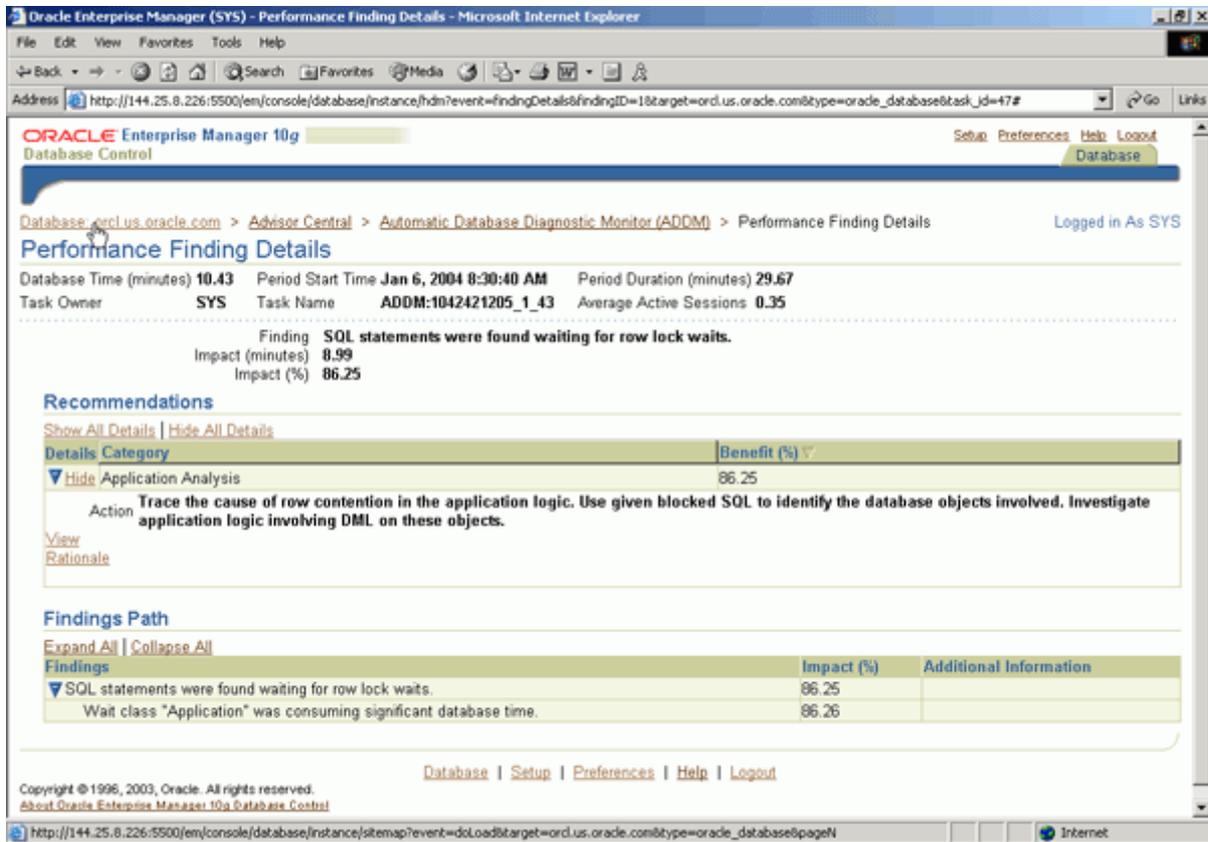
4. 您看到了解决该性能问题需要采取的操作。 单击 View Rationale 。
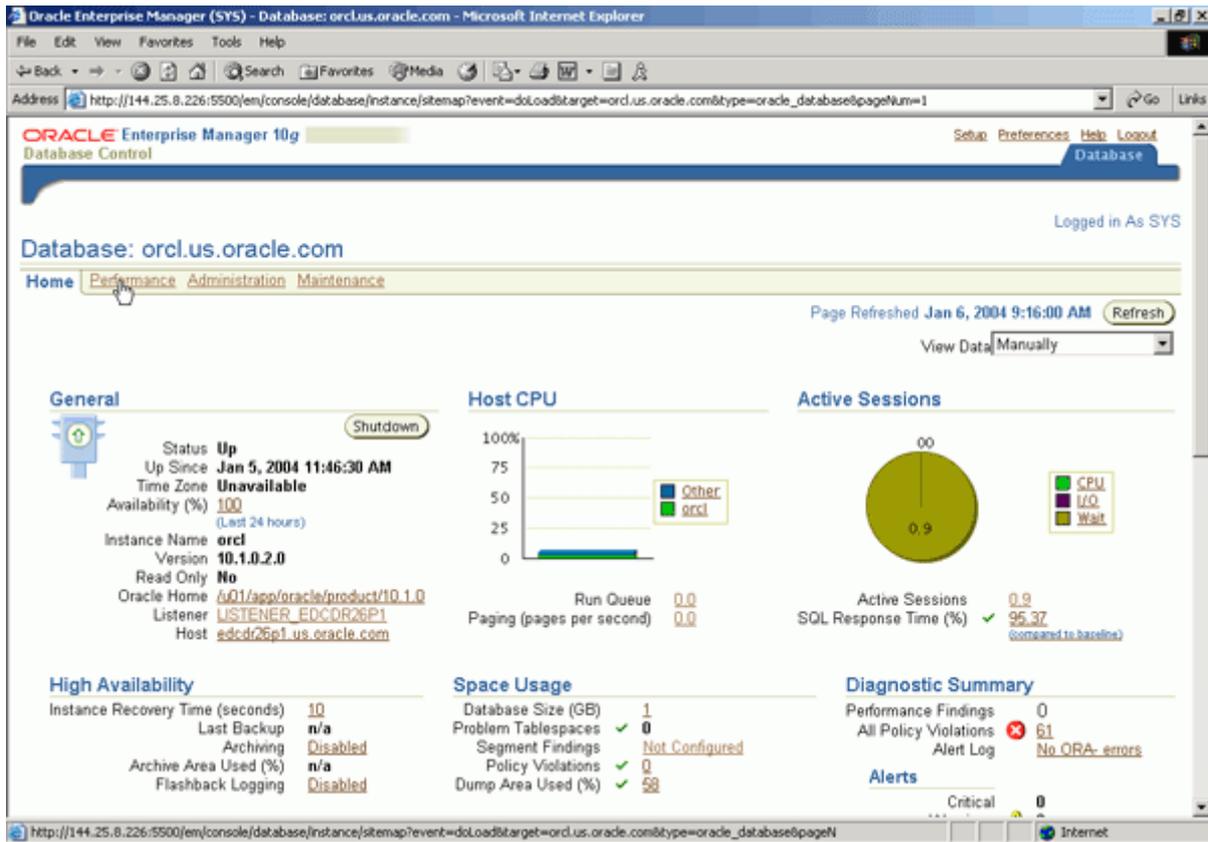
5. 您将看到引起问题的特定 SQL ID。 单击 Close 。
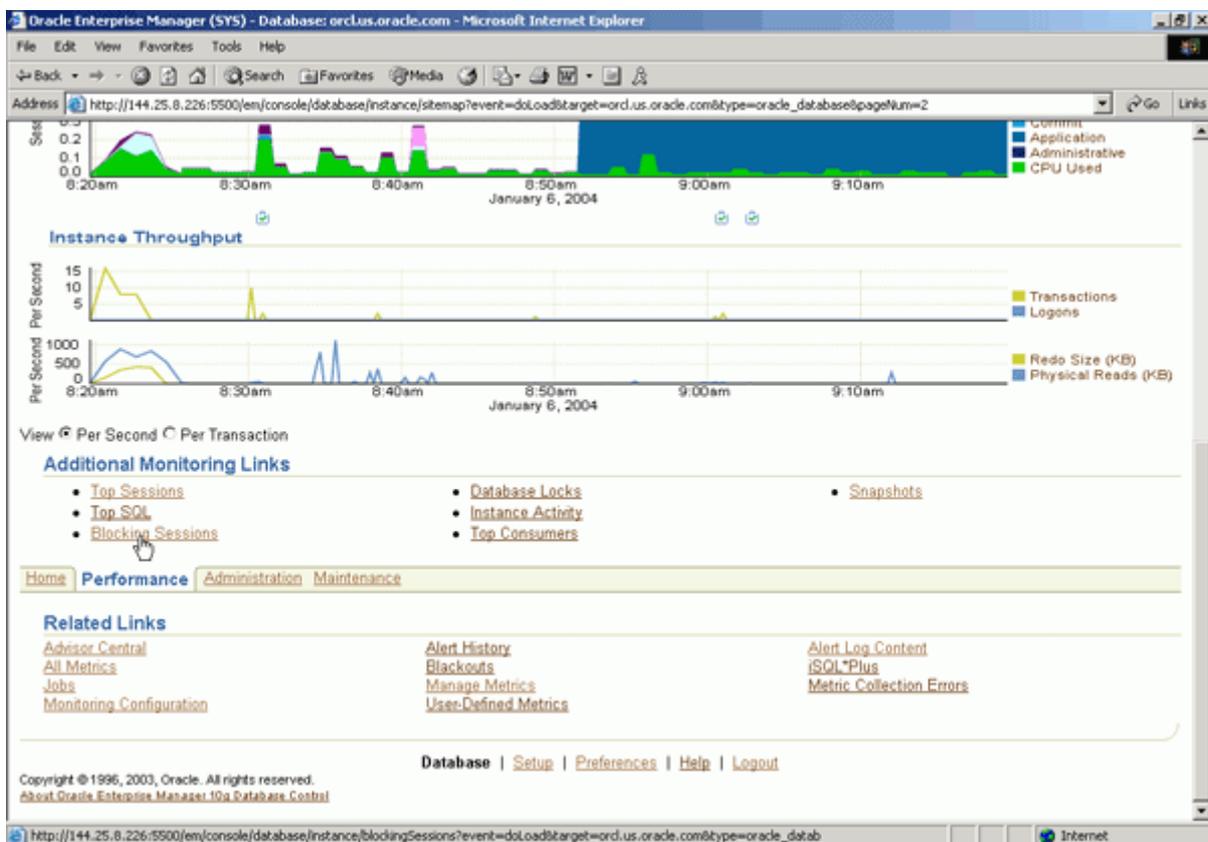
6. 要解决性能问题，请单击路径式导航栏中的 Database 导航项。

7. 单击 Performance 。

8. 向下滚动，然后选择 Additional Monitoring Links 下的 Blocking Sessions 。

9. 确保选择了最高级别的 HR，然后单击 Kill Session。

10. 单击 Yes ，终止会话。

11. 会话终止。 单击路径式导航栏中的 Database 导航项，然后单击 Home 。

12. 注意警报已消失。

使用 SQL Tuning Advisor

　　数据库管理员或操作员执行以下任务来响应性地监控并诊断数据库性能问题：

- 查看数据库等待
- 检查 Database Wait 类的 TOP SQL

- 利用 SQL 调整顾问工具调整 SQL 语句
- 查看 SQL 语句的 SQL 执行细节

查看数据库等待

　　您将首先启动几个工作负载会话。然后您将查看数据库工作负载。创建一个名称为 $HOME/wkdir 的目录。下载 perflab.tar 文件，并将文件解压缩至 $HOME/wkdir 目录中。 执行以下步骤：

1. 打开一个命令行窗口，然后运行以下 OS 脚本：

./ setup_perflab.sh

注意： 这个脚本运行将花费大约 4 分钟。

2. 回切到 Enterprise Manager 窗口。 选择 Administration 链接。

3. 在标题为 Workload 的部分中，单击 Automatic Workload Repository 链接。

4. 确定已经为该数据库采集了多少快照。 查看 Snapshots 下的数量和最后一次 ADDM 快照采集的时间。 应该有至少三个快照。 单击路径式导航栏中的 Database 导航项。
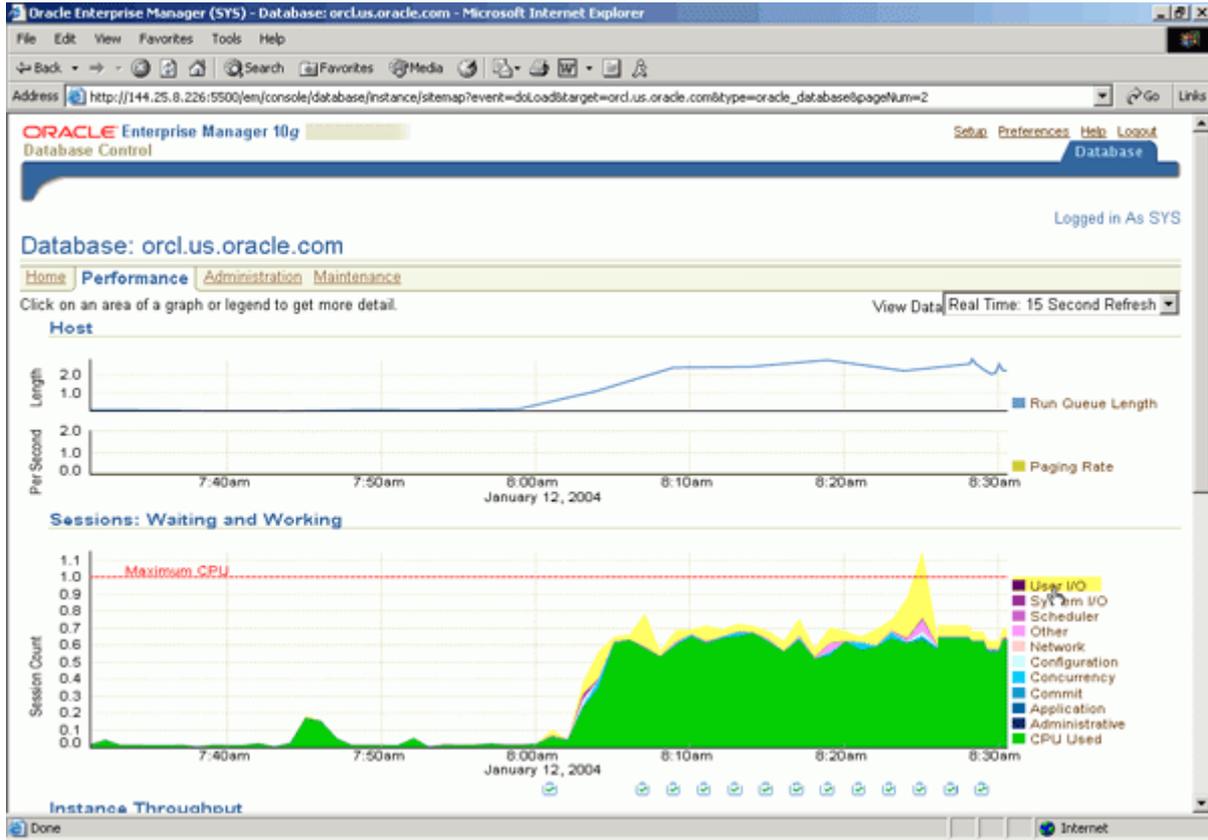
5. 单击 Performance 链接。
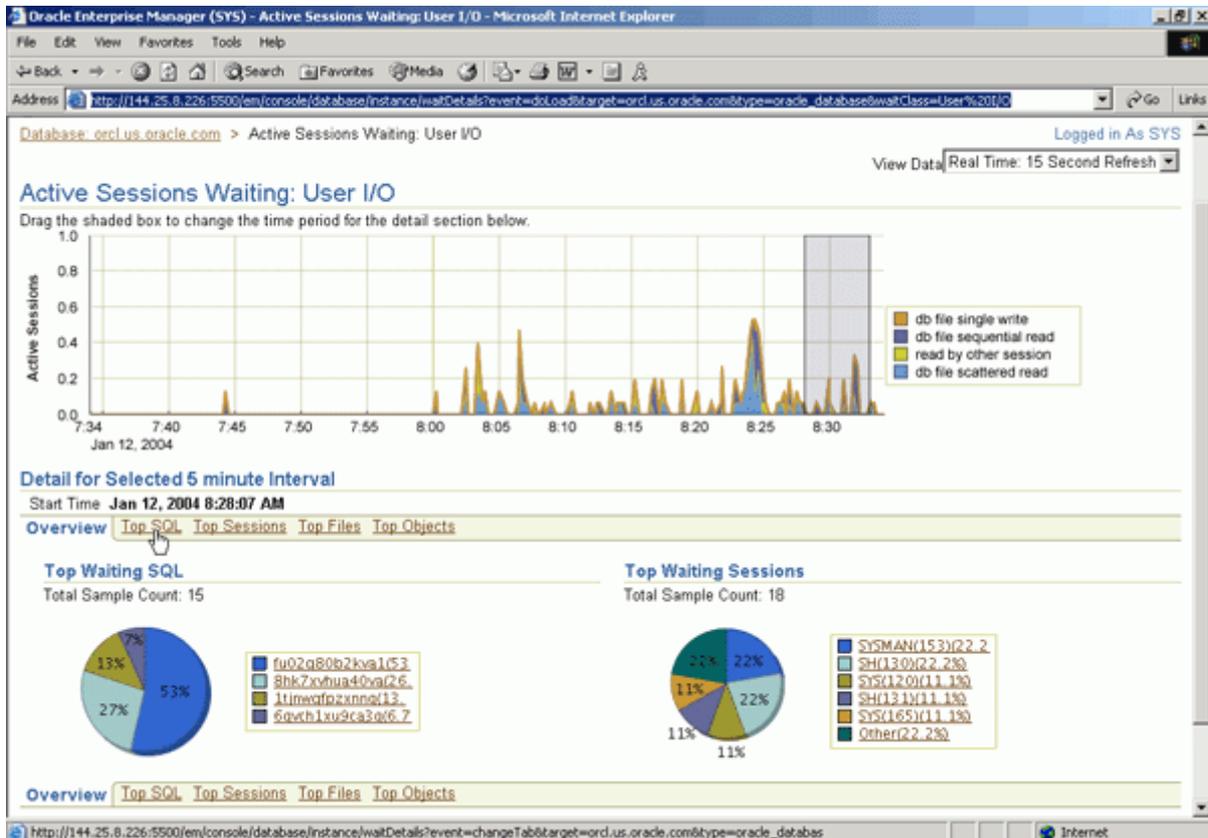
6. 在 Performance 窗口中查看 Sessions: Waiting and Working 图。 显示一个表示数据库当前工作负载的图表。（向图表填充数据可能要花一分钟）用 ADDM 快照采集的数据来填充这个图表。 图表的边上是图例。 每一个图例项目具有不同的颜色。 您可以从图中快速确定占用最多时间的项目是黄色的，即用户 I/O。单击 User I/O 链

接。
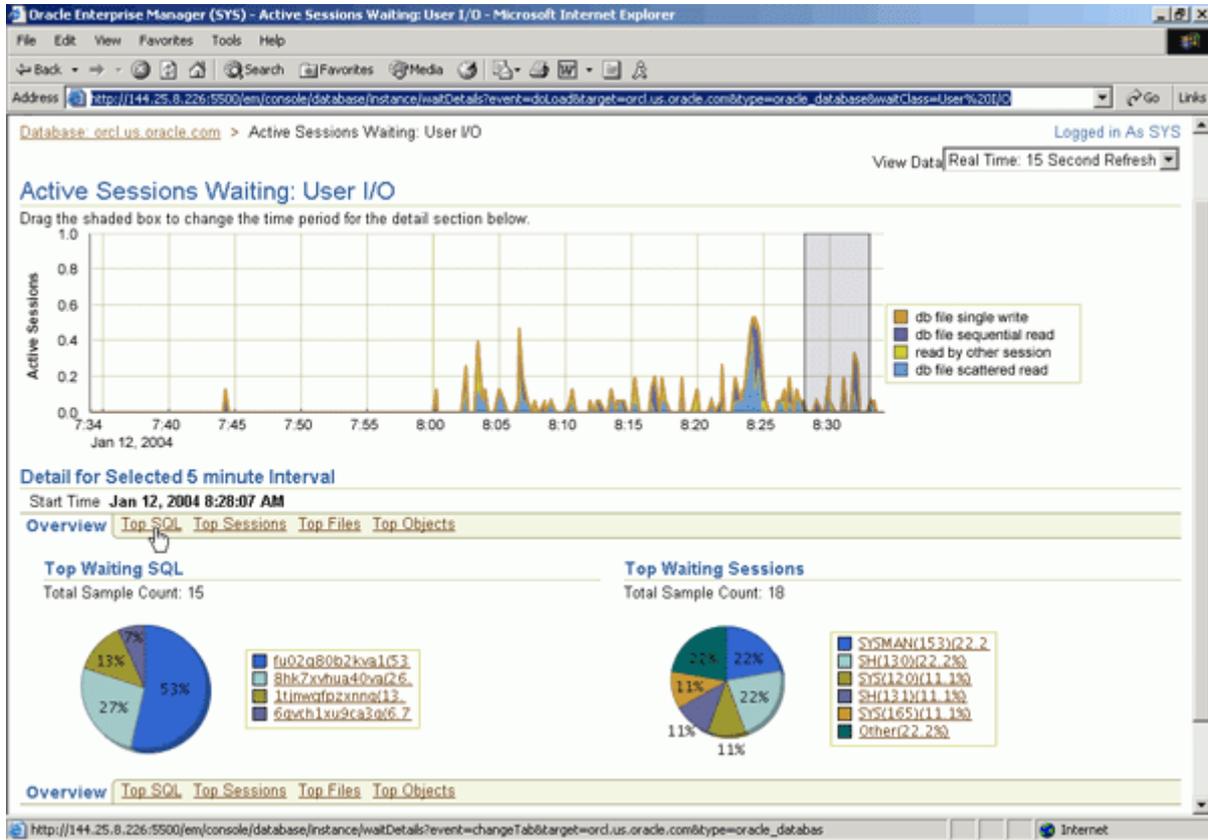
7. 在 Active Sessions Waiting: User I/O 图表下有两个饼图。 查看左边的饼图 Top Waiting SQL。 这显示占绝大多数的等待 (53%) 是由一条 SQL 语句造成的。 查看右边的饼图 Top Waiting Sessions。 这个饼图显示当前最活跃的几个会话等待的时间几乎占相同的百分比。
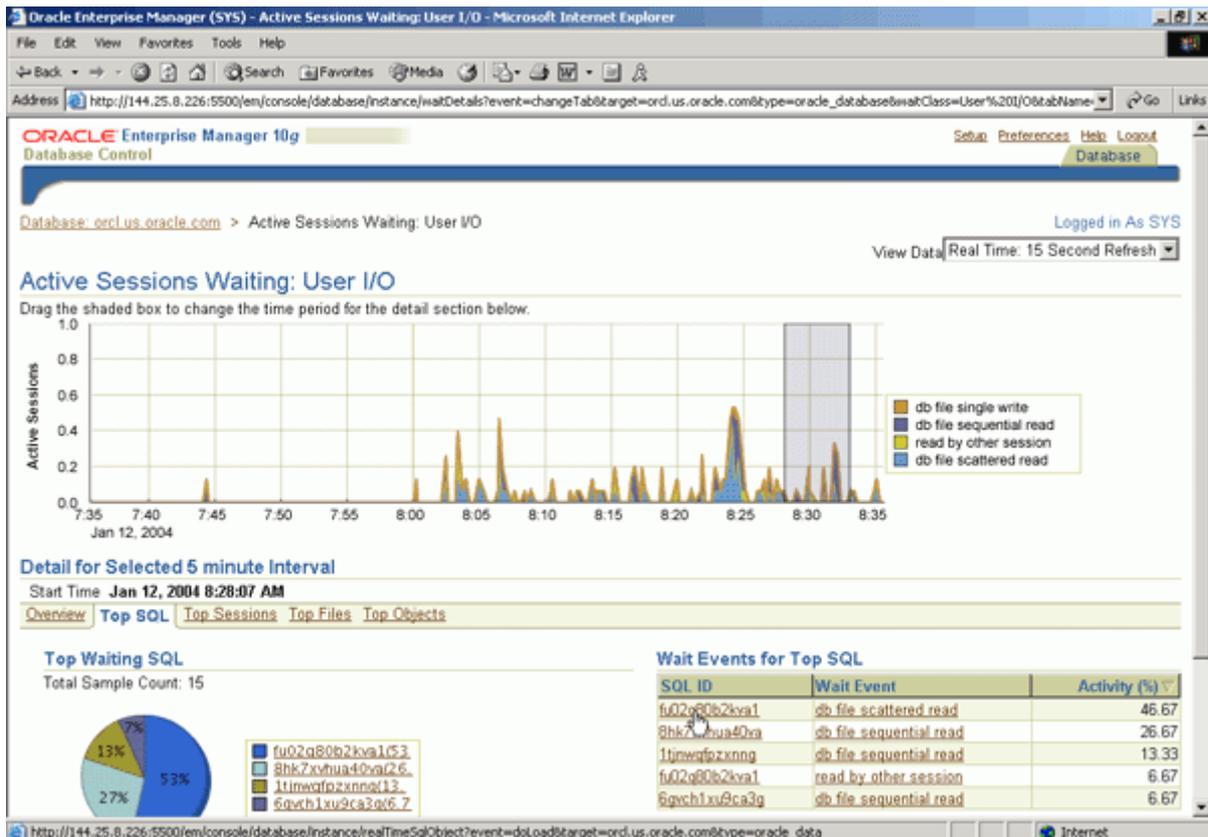
检查 Database Wait 类的 TOP SQL

如之前的任务所示，有一条 SQL 语句导致了大部分的数据库等待。 在这个任务中，您将深入了解根本原因。 执行以下操作：
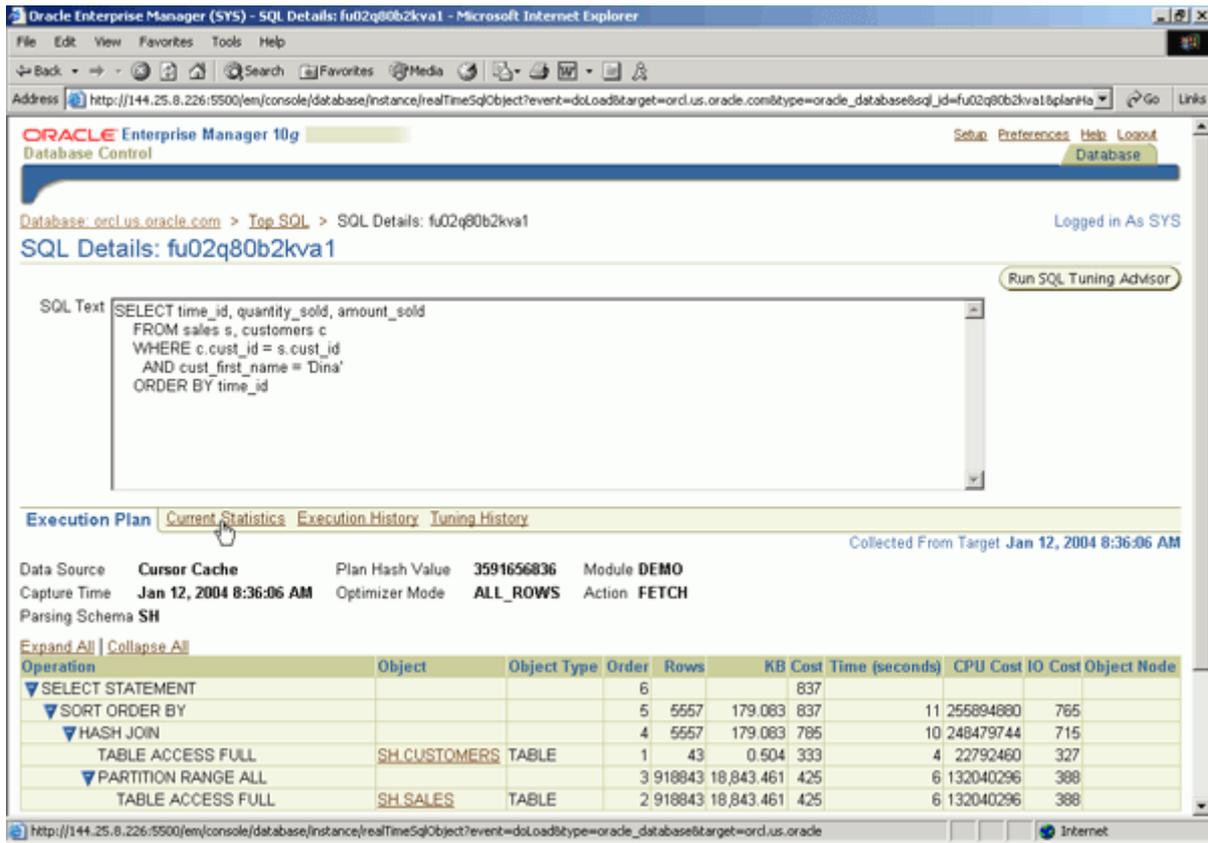
1. 从 Active Sessions Waiting: User I/O 页面中，单击页面中部的 Top SQL 。
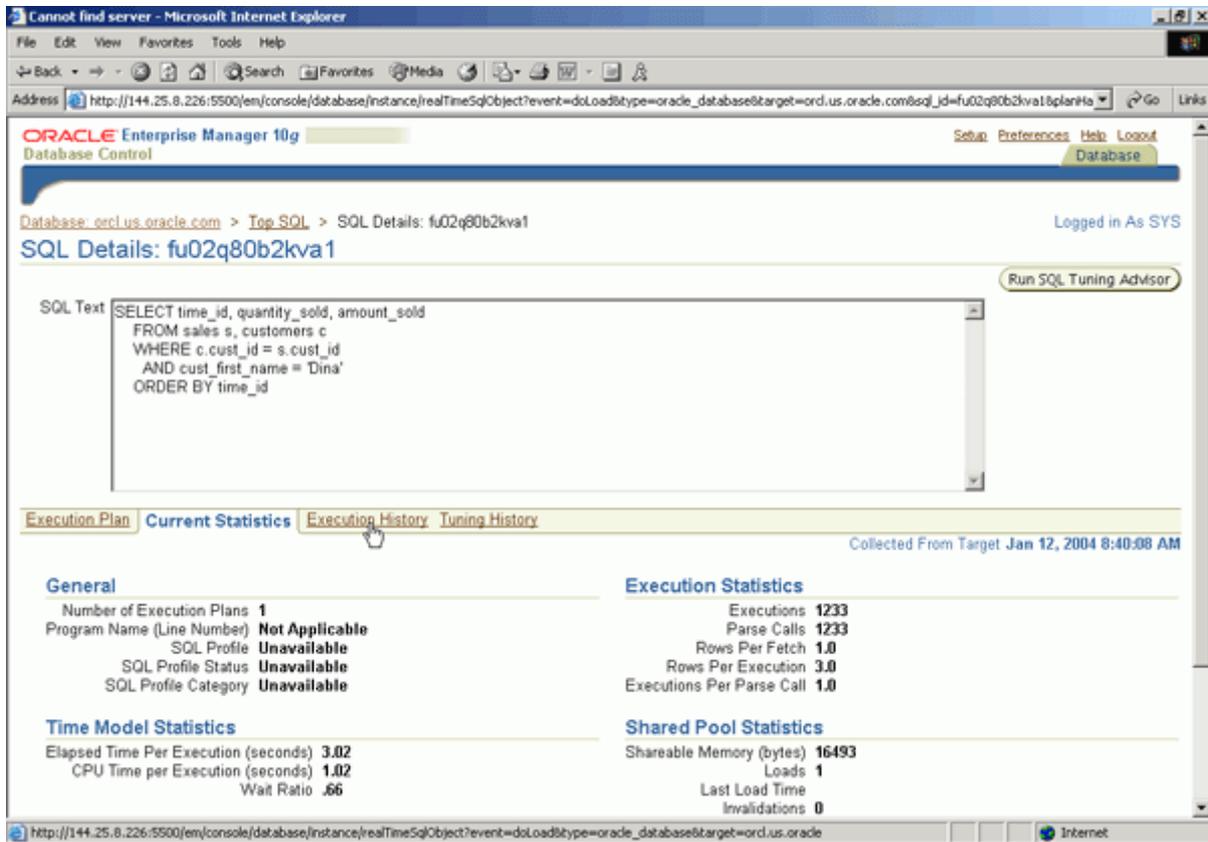
2. 在出现的详细页面上，查看 Wait Events for Top SQL 表，该表按 活动（%） 排序。 您可以看到 Top SQL 语句将其大部分时间花费在了 'db file scattered read' 活动上。 单击具有最高活动百分比的 SQL 语句的 SQL ID。
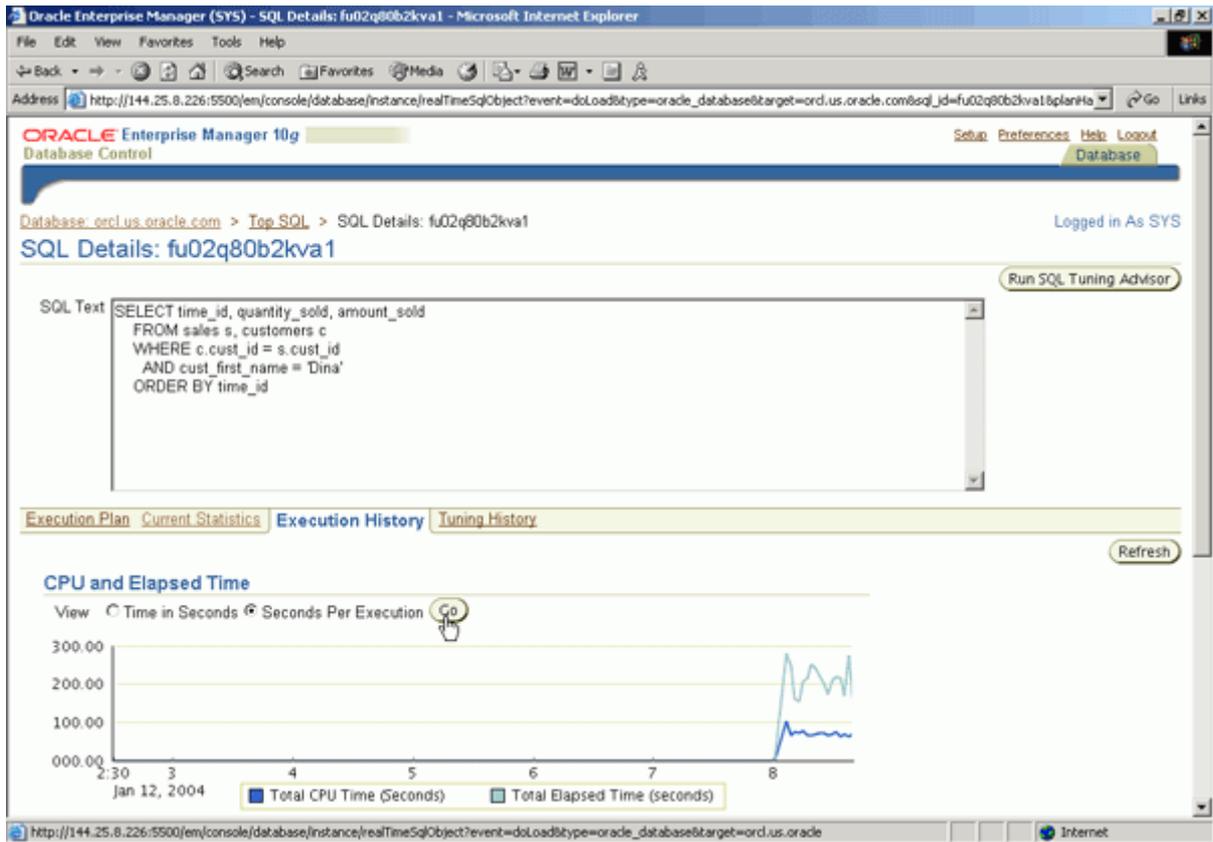
3. 显示这条 SQL 语句的执行计划。 单击 Current Statistics 标签。
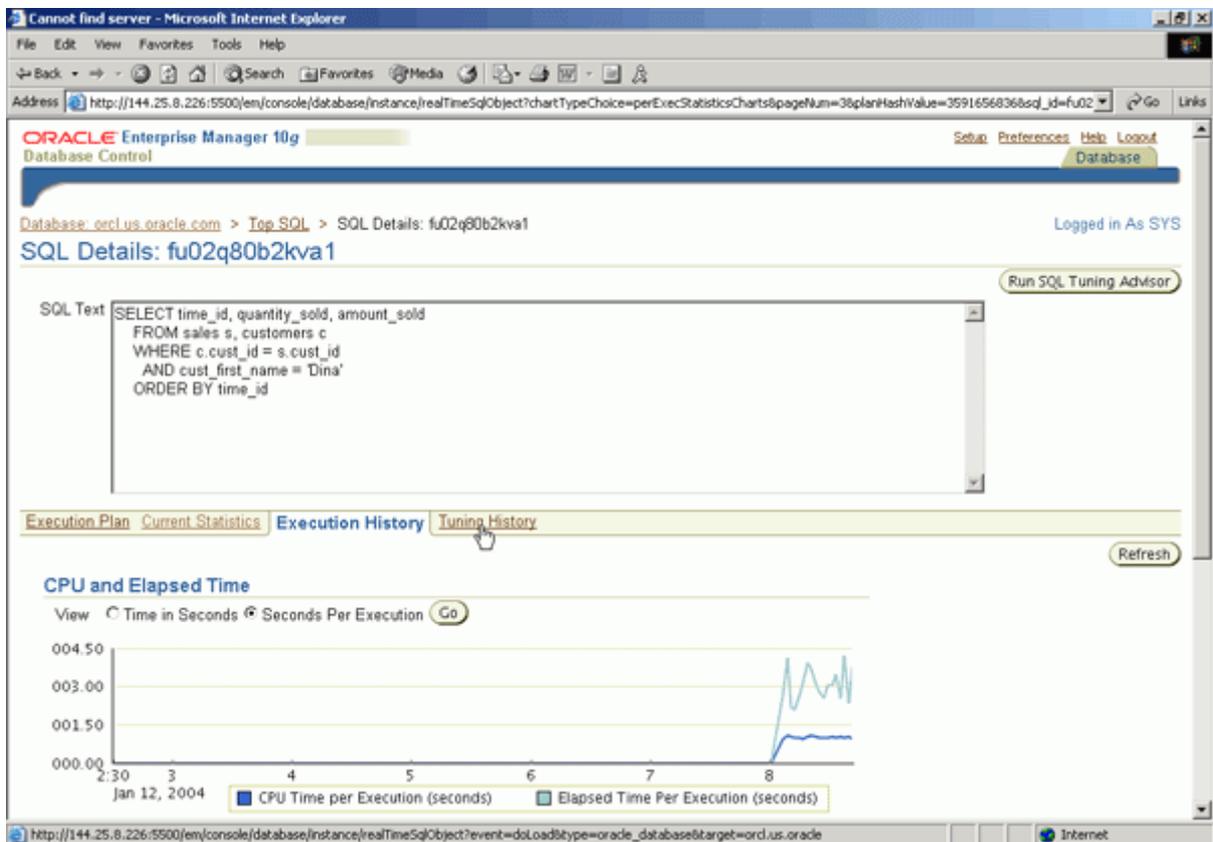
4. 显示这条 SQL 语句的统计数据。 单击 Execution History 标签。
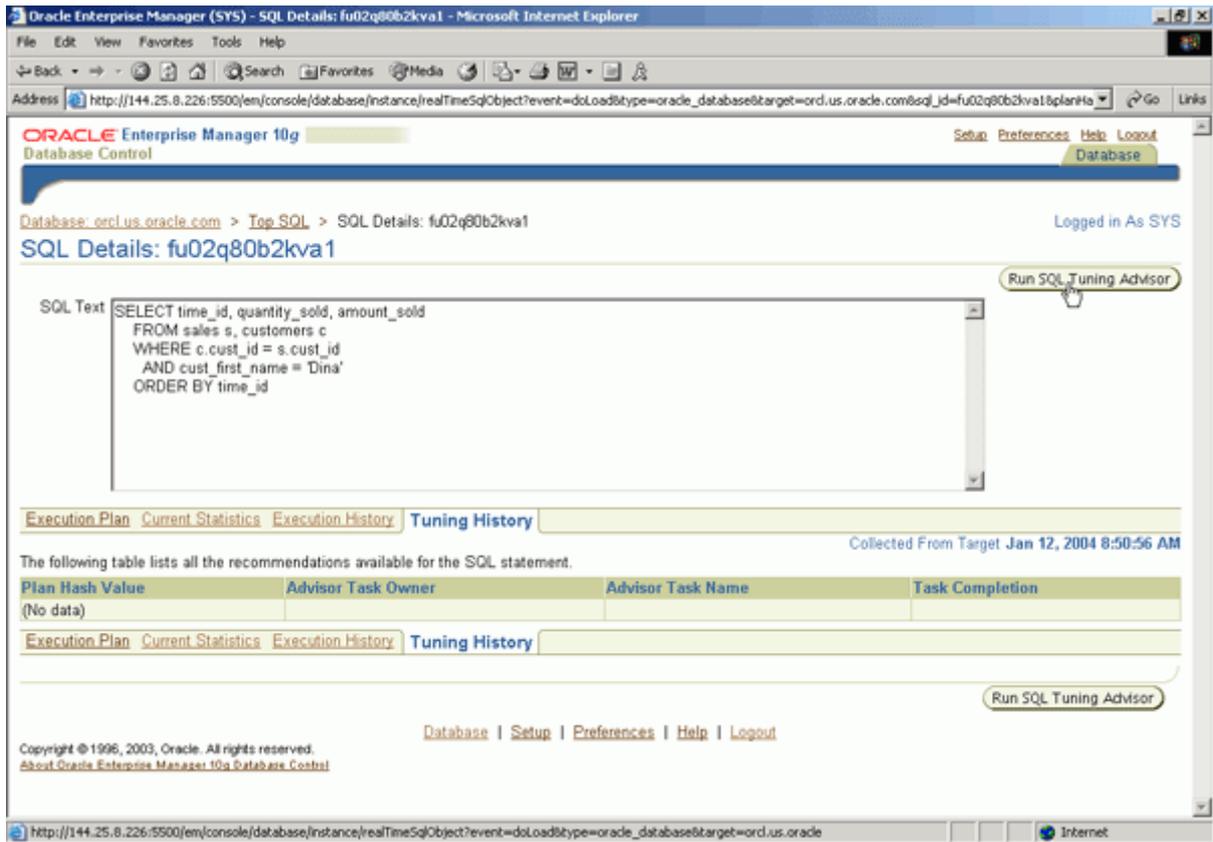
5. 显示这条 SQL 语句的统计分析图表。CPU 和 Elapsed Time 图表显示在一段时间上执行这条 SQL 语句的 CPU 使用率。 选择 Seconds Per Execution ，单击 Go ，显示这条 SQL 语句的每次执行使用的时间和资源。

6. 从显示的图表中，可以确定 CPU 资源使用率在增加，执行这条 SQL 语句花费的时间也在增加。 单击 Tuning History 标签。
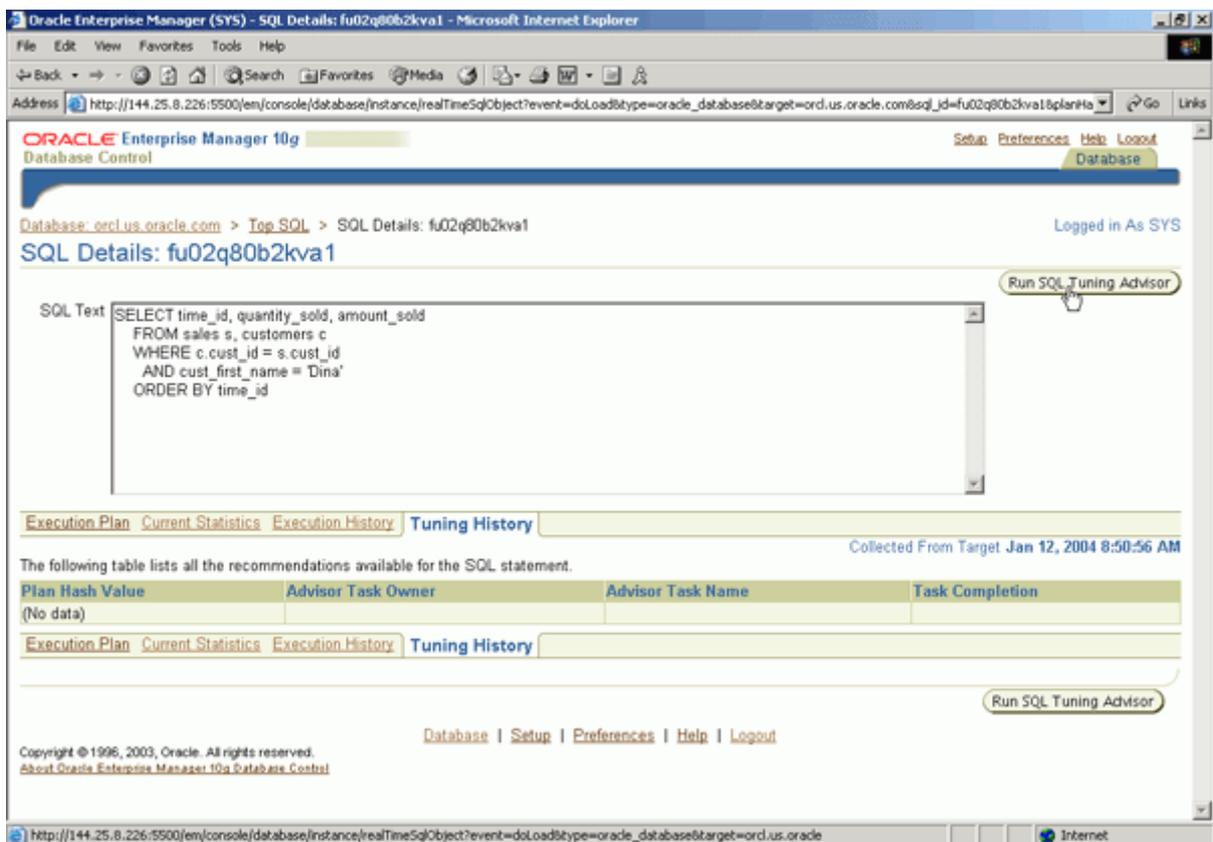
7. 显示之前为这条 SQL 语句提供的调整建议。 当前没有建议。 您现在准备好了使用 SQL Tuning Advisor 来调整 SQL 语句。
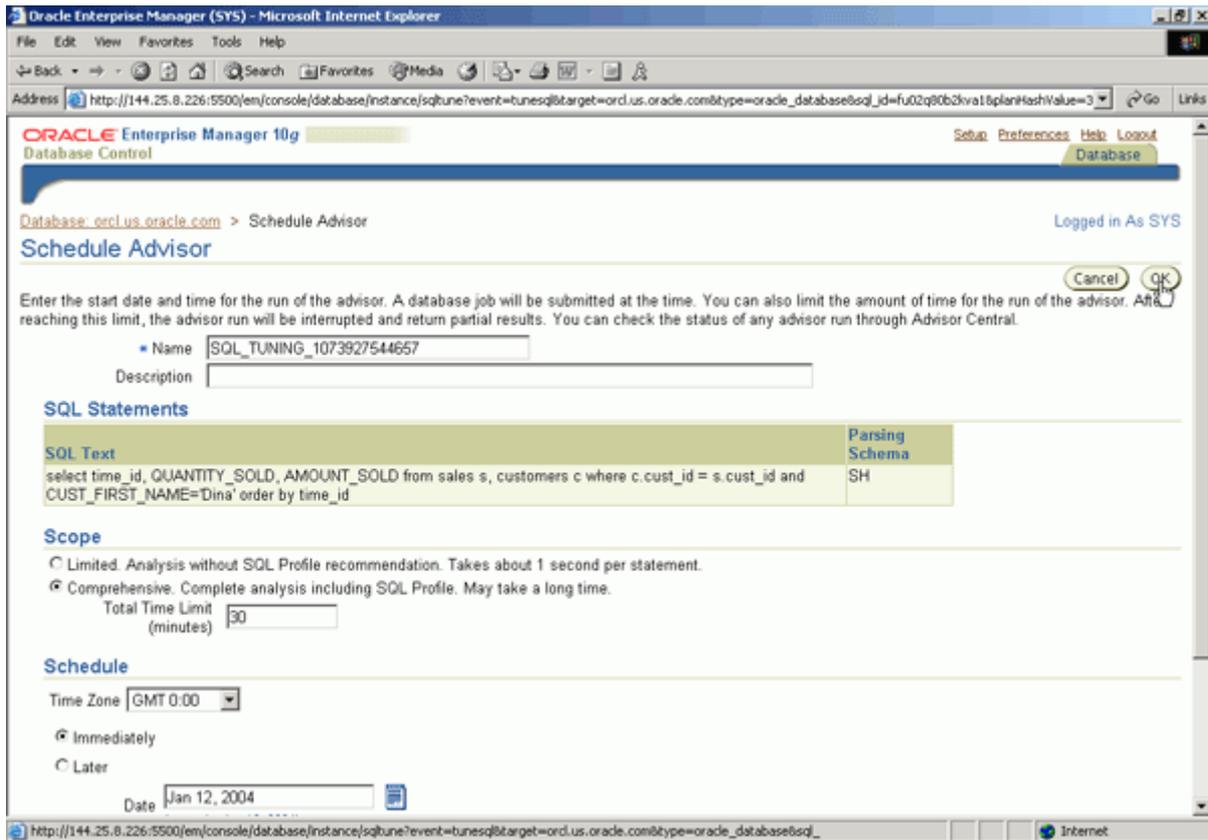
利用 SQL 调整顾问工具调整 SQL 语句

正如在上一个关于响应式调整的部分中确定的那样，目标 SQL 语句需要调整。 SQL 调整顾问工具将为您调整执行计划。 执行以下操作:
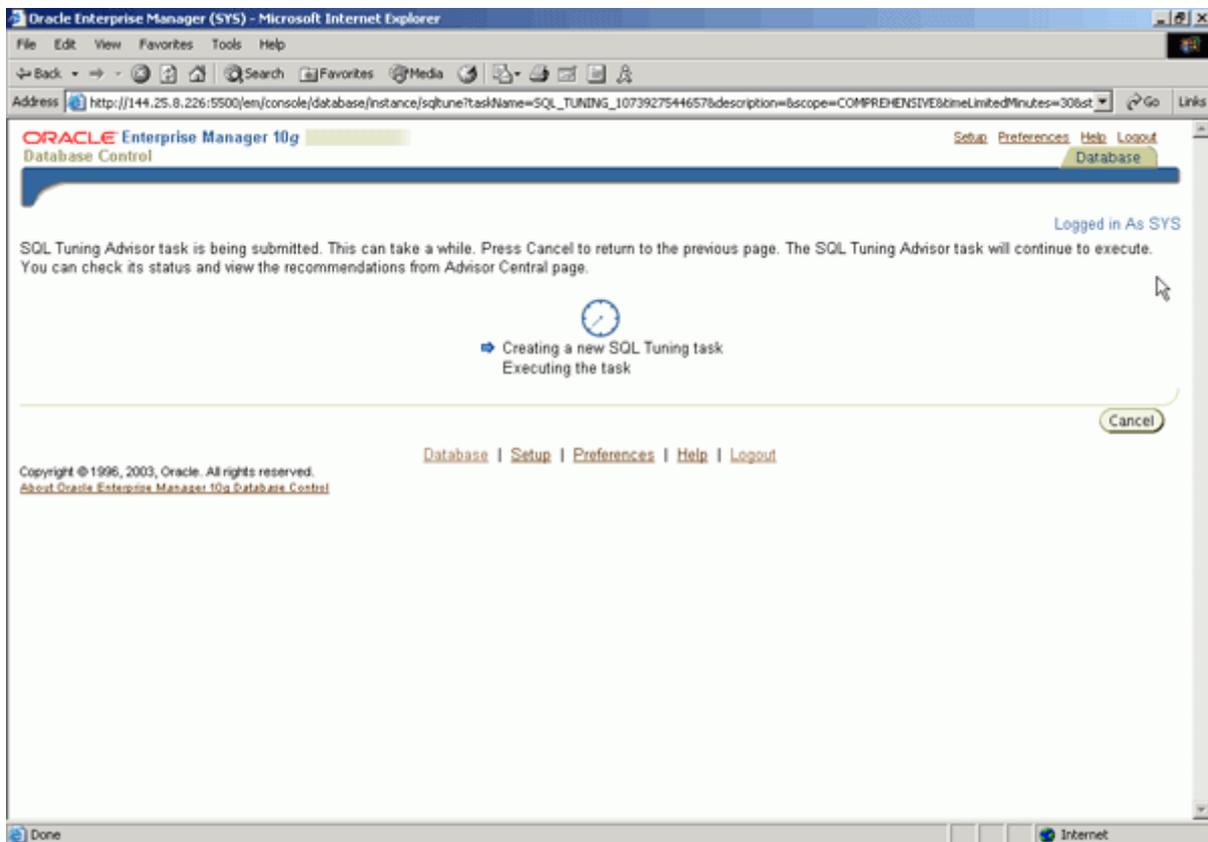
1. 单击 Run SQL Tuning Advisor 。
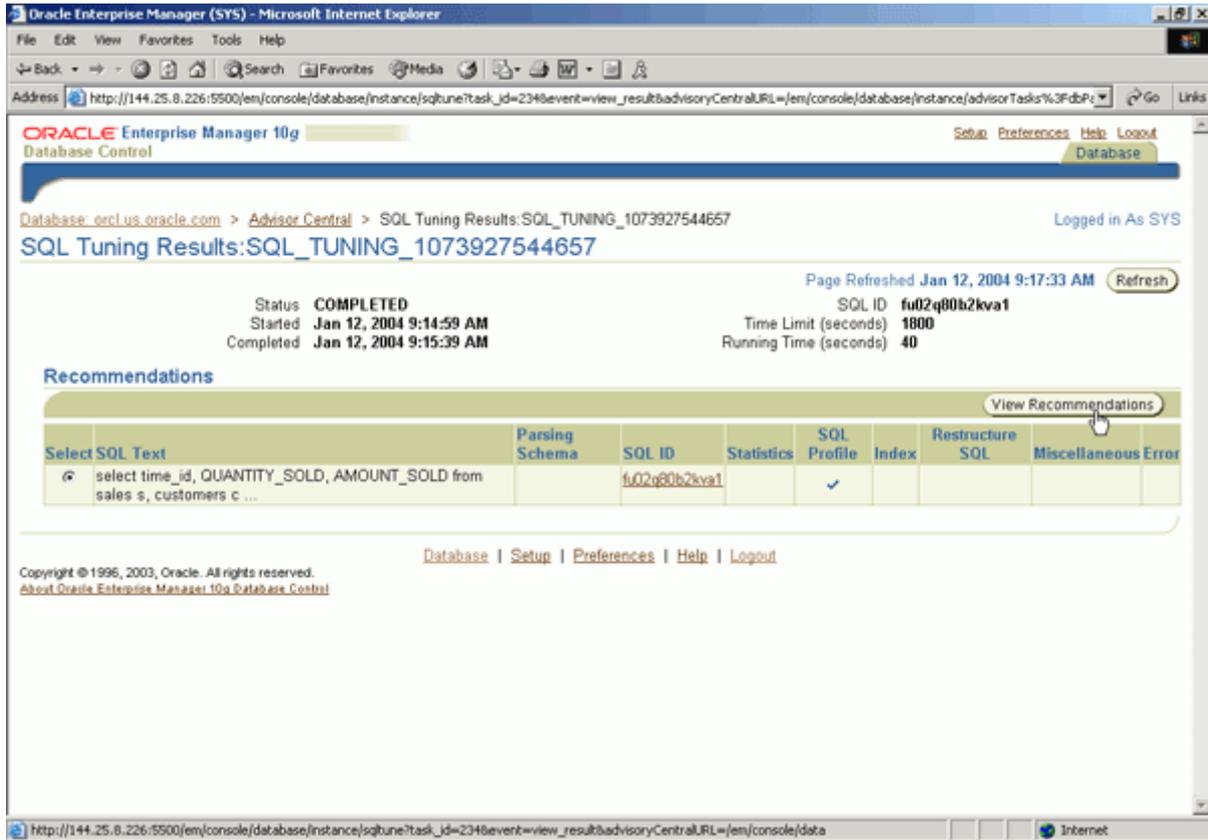
2. 在 Schedule Advisor 窗口中，确保选择了 Scope 下的 Comprehensive ，作业将 立即 安排。单击 OK 。
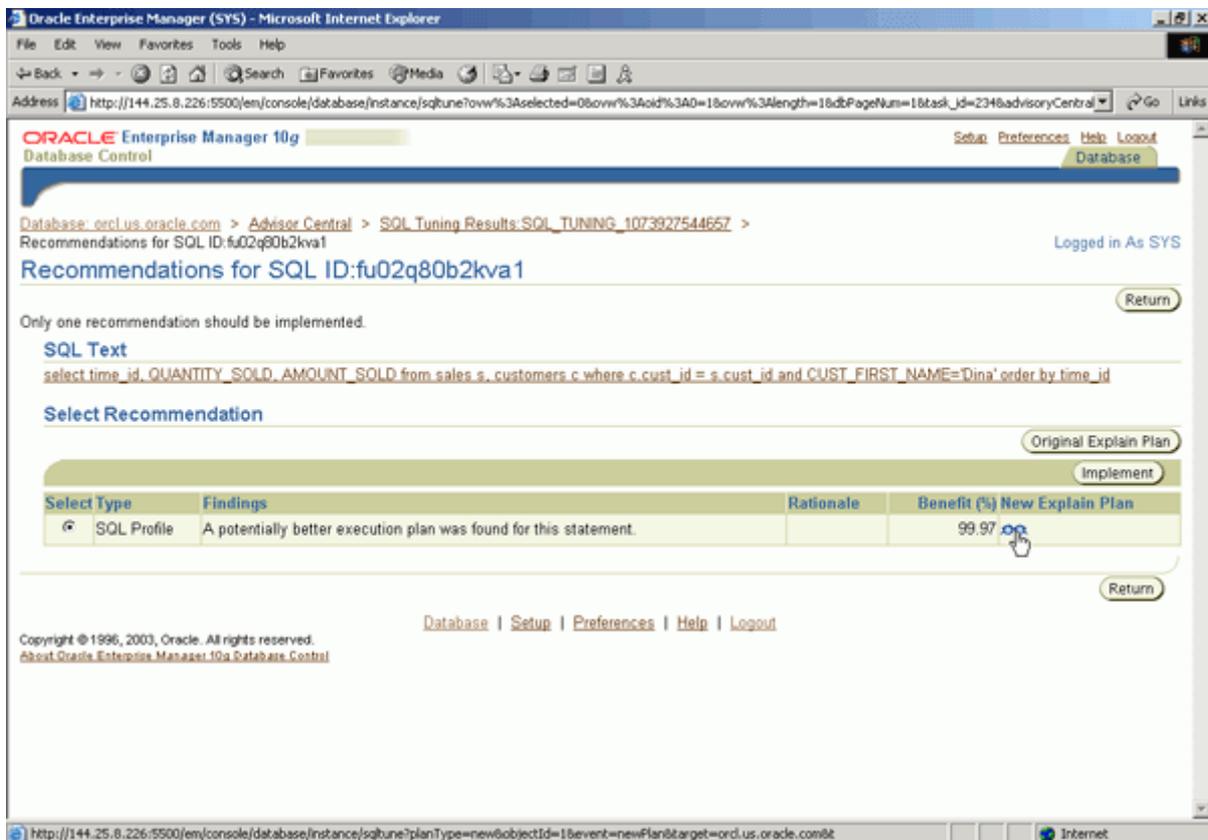
3. SQL Tuning Advisor 将创建一个任务，以分析 SQL 语句，任务一旦完成，将显示一组调整建议。
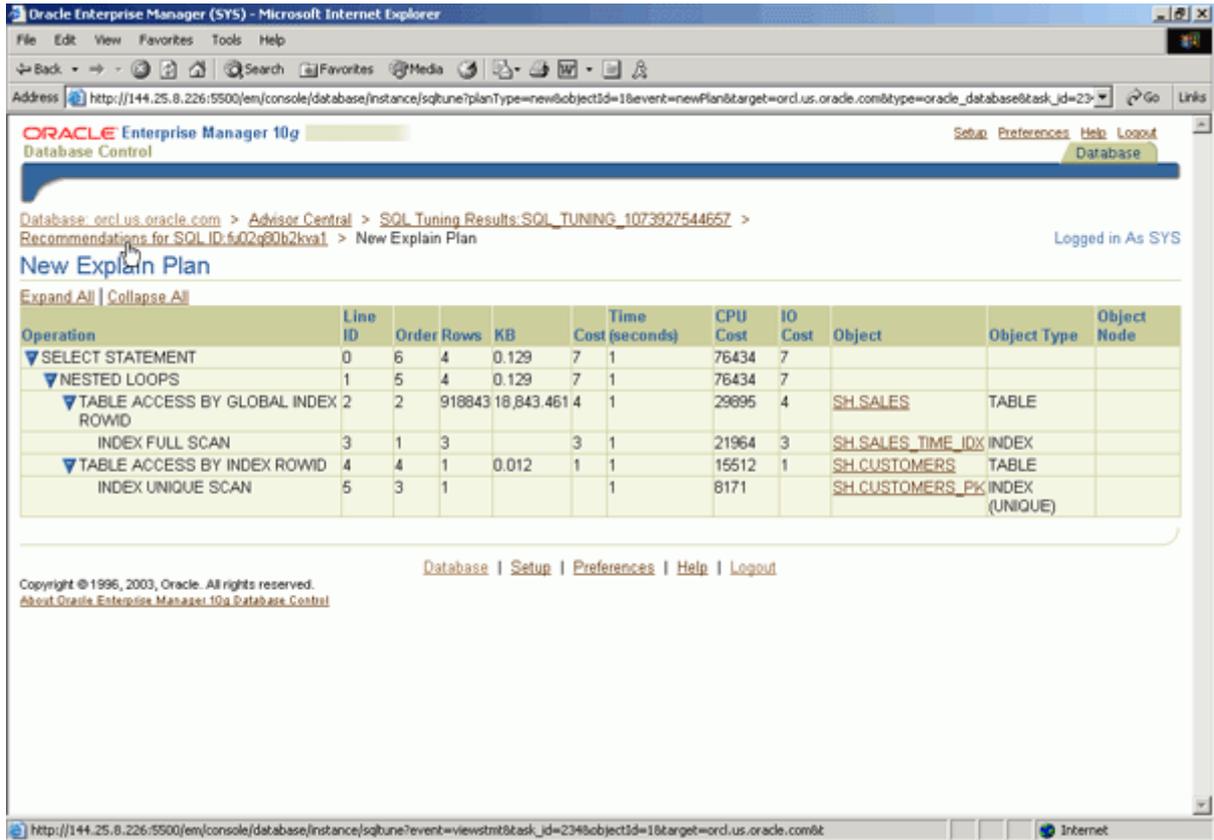
4. 单击 View Recommendations 。
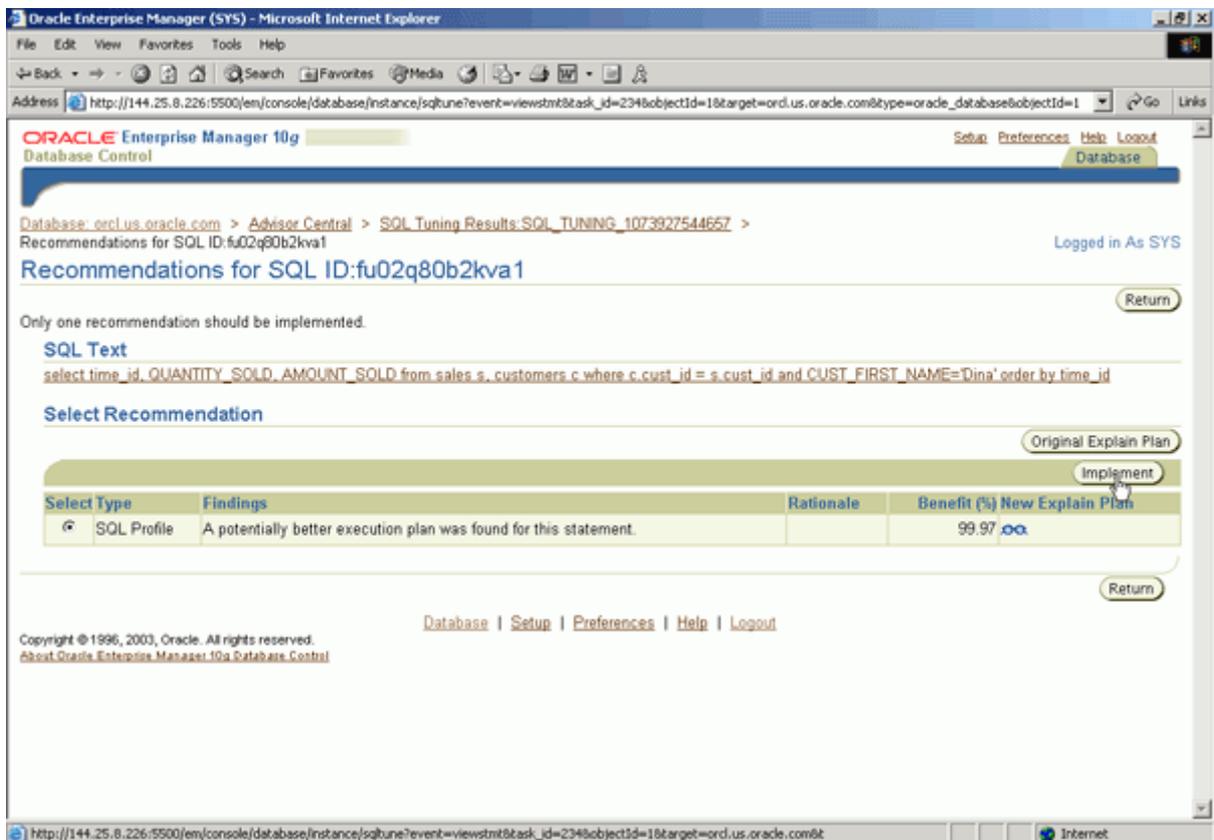
5. 单击 New Explain Plan 按钮，查看建议的修改。
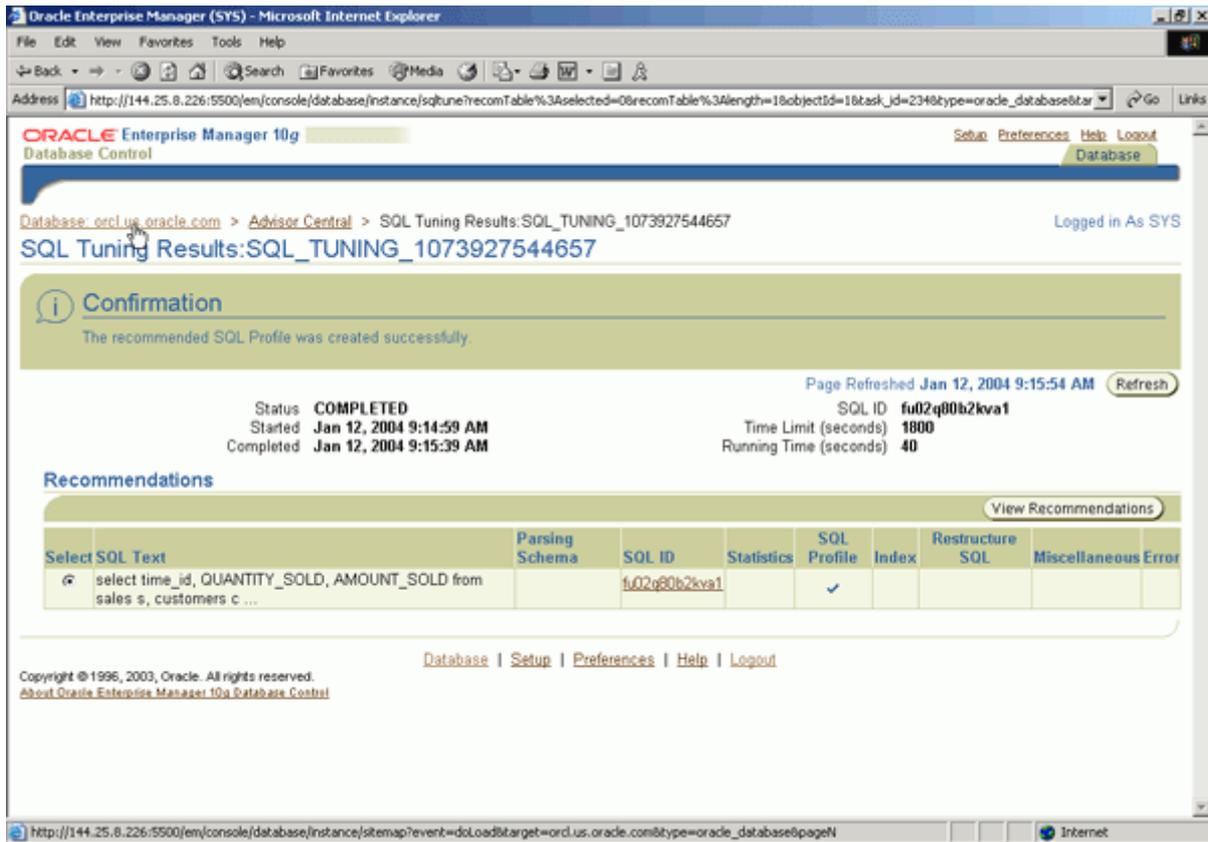
6. 正如您所看到的，新的解释计划删除了全部的表扫描。 单击浏览器后退按钮，返回上一页。

7. 单击 Implement 按钮，执行调整建议。
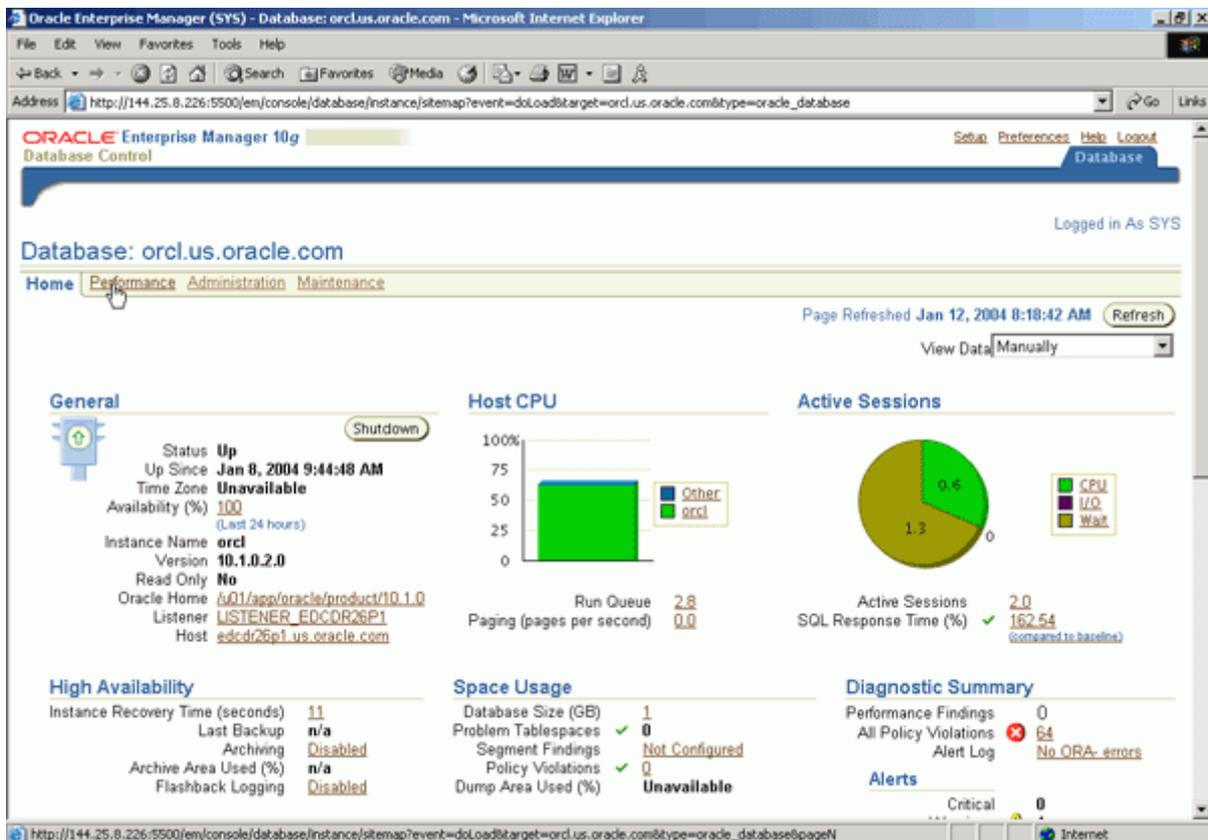
8. 出现一条确认页面，指示 SQL Profile 成功创建。 单击路径式导航栏中的 Database 导航项。

查看 SQL 语句的 SQL 执行细节

您已执行了调整建议，现在查看 SQL 语句和其执行细节。 执行以下步骤。

1. 执行 Performance 。

2. 向下滚动至 Sessions: Waiting and Working 图表。 等待一分钟，观察用户 I/O 如何减少。

使用 SQL Access Advisor

SQL Access Advisor 提供了许多过程，可以调用这些过程来帮助确定创建和丢弃哪些实体化化视图和索引。它利用基于您的模式的，假定的工作负载，或从实际的工作负载（可能由用户提供、来自 Oracle Trace 或来自 SQL 高速缓存的内容）中作出决策。

还可以根据不同的标准对工作负载进行过滤，如只使用包含这些表的查询，或在这个范围中有优先权的查询。
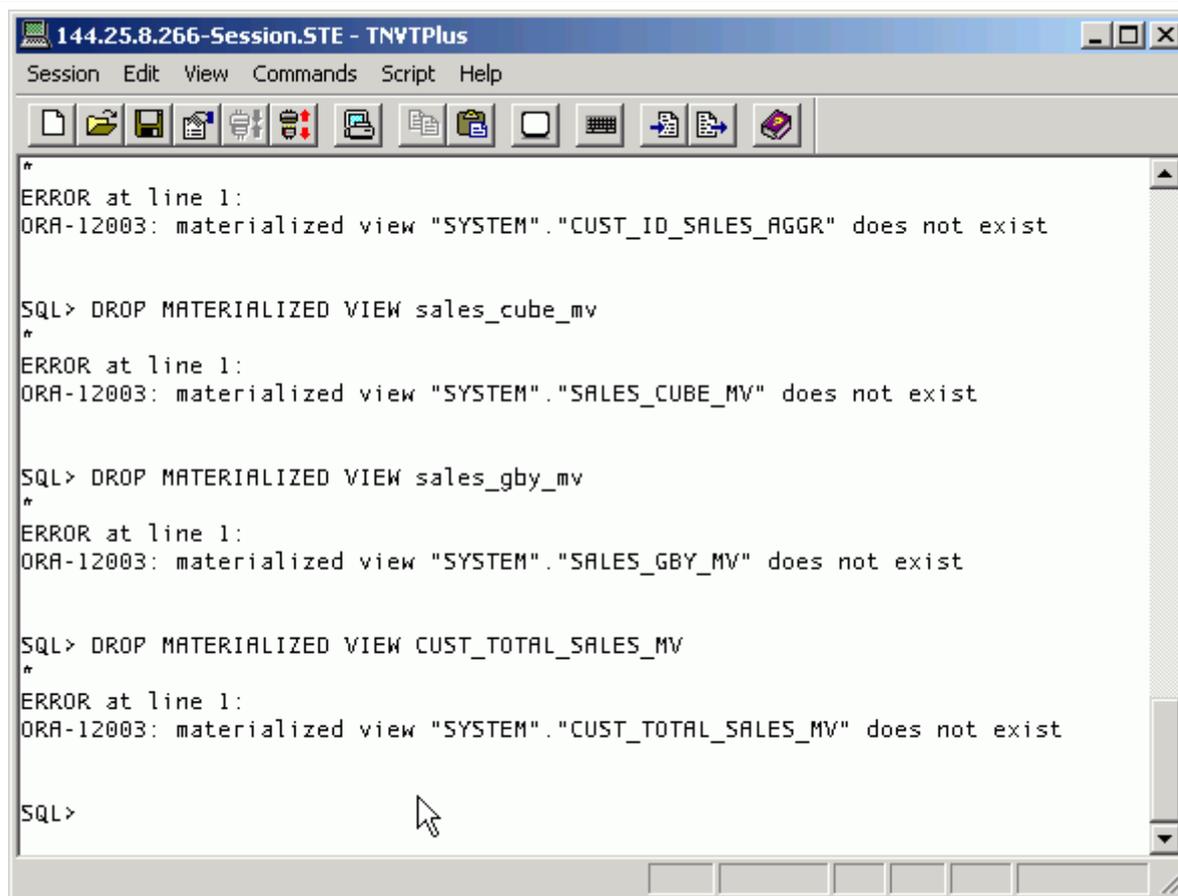
- 准备环境
- 利用 SQL 高速缓存获取建议
- 查看和执行建议

准备环境

执行以下步骤，为使用 SQL Access Advisor 准备好环境。 当 Advisor 在运行时，实体化视图和索引可能存在，但为了举例的目的，把它们删除了，以便您可以看到 Advisor 的建议。 您还需要设置高速缓存，以便 SQL Access Advisor 能够生成建议。 执行以下操作：

1. 打开一个终端窗口，然后执行以下命令，清理环境：

sqlplus system/ DROP MATERIALIZED VIEW all_cust_sales_mv;

DROP MATERIALIZED VIEW costs_mv;

DROP MATERIALIZED VIEW costs_pm_mv;

DROP MATERIALIZED VIEW cust_sales_mv;

DROP MATERIALIZED VIEW some_cust_sales_mv;

DROP MATERIALIZED VIEW cust_id_sales_aggr;

DROP MATERIALIZED VIEW sales_cube_mv;

DROP MATERIALIZED VIEW sales_gby_mv;

DROP MATERIALIZED VIEW CUST_TOTAL_SALES_MV;

DROP MATERIALIZED VIEW CUST_SALES_TIME_MV;

```
144.25.8.266-Session.STE - TNYTPlus                          _ □ ×

Session  Edit  View  Commands  Script  Help

 D | ☞ | ⊟ | 🖺 | 🖳 | 🖳 | ⊞ | 🖺 | 🖺 | □ | ▦ | 🔁 | 🔁 | ◆

*
ERROR at line 1:
ORA-12003: materialized view "SYSTEM"."CUST_ID_SALES_AGGR" does not exist


SQL> DROP MATERIALIZED VIEW sales_cube_mv
*
ERROR at line 1:
ORA-12003: materialized view "SYSTEM"."SALES_CUBE_MV" does not exist


SQL> DROP MATERIALIZED VIEW sales_gby_mv
*
ERROR at line 1:
ORA-12003: materialized view "SYSTEM"."SALES_GBY_MV" does not exist


SQL> DROP MATERIALIZED VIEW CUST_TOTAL_SALES_MV
*
ERROR at line 1:
ORA-12003: materialized view "SYSTEM"."CUST_TOTAL_SALES_MV" does not exist


SQL>
```

2. 现在您需要创建高速缓存。执行下列命令：

alter system flush shared_pool;

grant advisor to sh;

connect sh/sh;

SELECT c.cust_last_name, sum(s.amount_sold) AS dollars,

sum(s.quantity_sold) as quantity

FROM sales s , customers c, products p

WHERE c.cust_id = s.cust_id

AND s.prod_id = p.prod_id
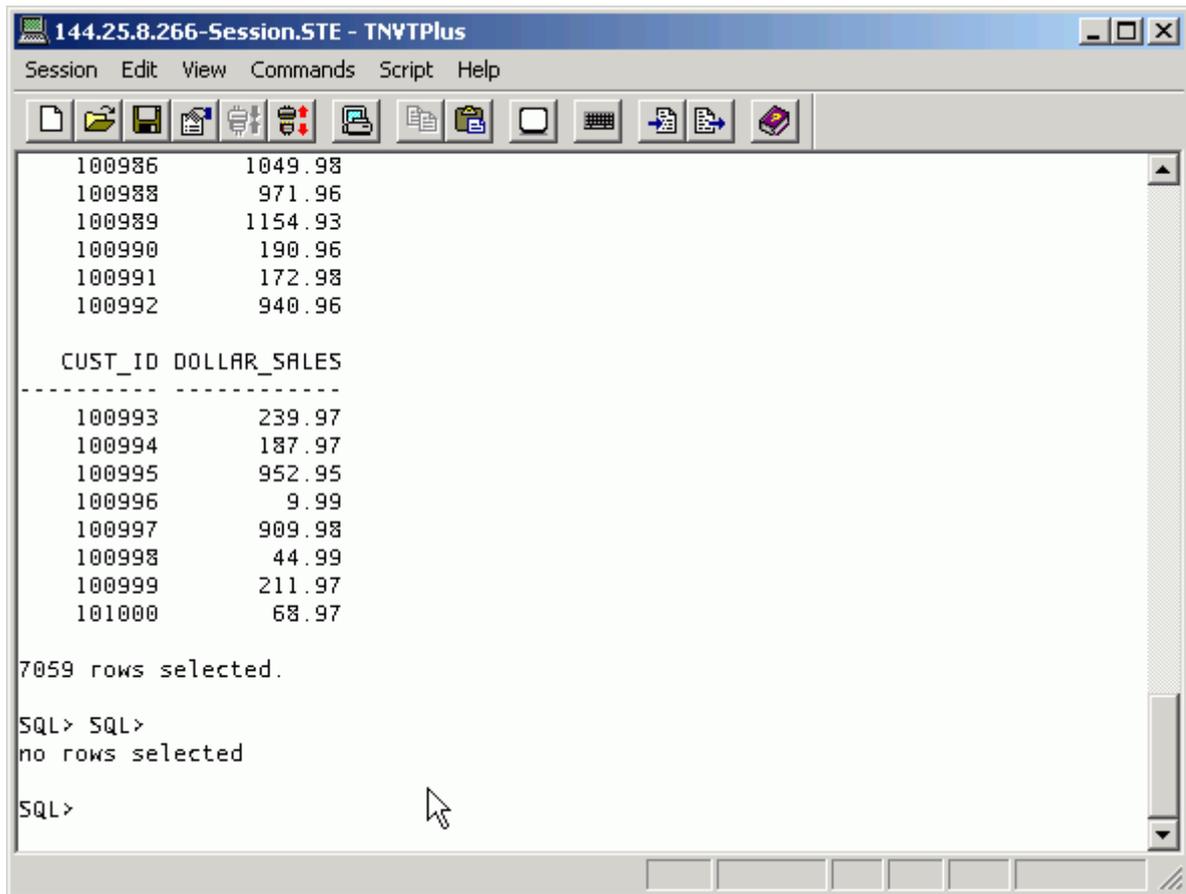
AND c.cust_state_province IN ('Dublin','Galway')

GROUP BY c.cust_last_name;

SELECT c.cust_id, SUM(amount_sold) AS dollar_sales

FROM sales s, customers c WHERE s.cust_id= c.cust_id GROUP BY c.cust_id;

select sum(unit_cost) from costs group by prod_id;
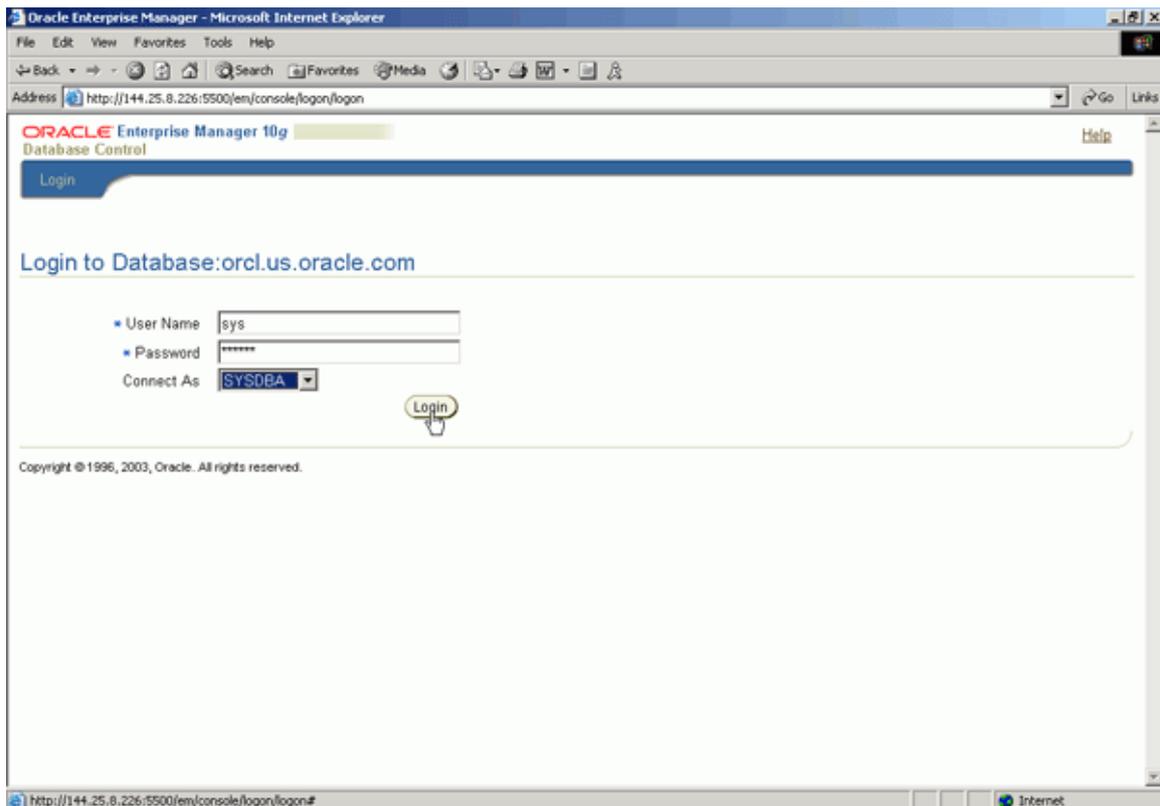
利用 SQL 高速缓存获取建议

    您将使用您刚刚设置的 SQL 高速缓存来获取来自 SQL Access Advisor 的建议。 执行以下操作：

1. 打开浏览器，输入以下 URL：

http:// :5500/em
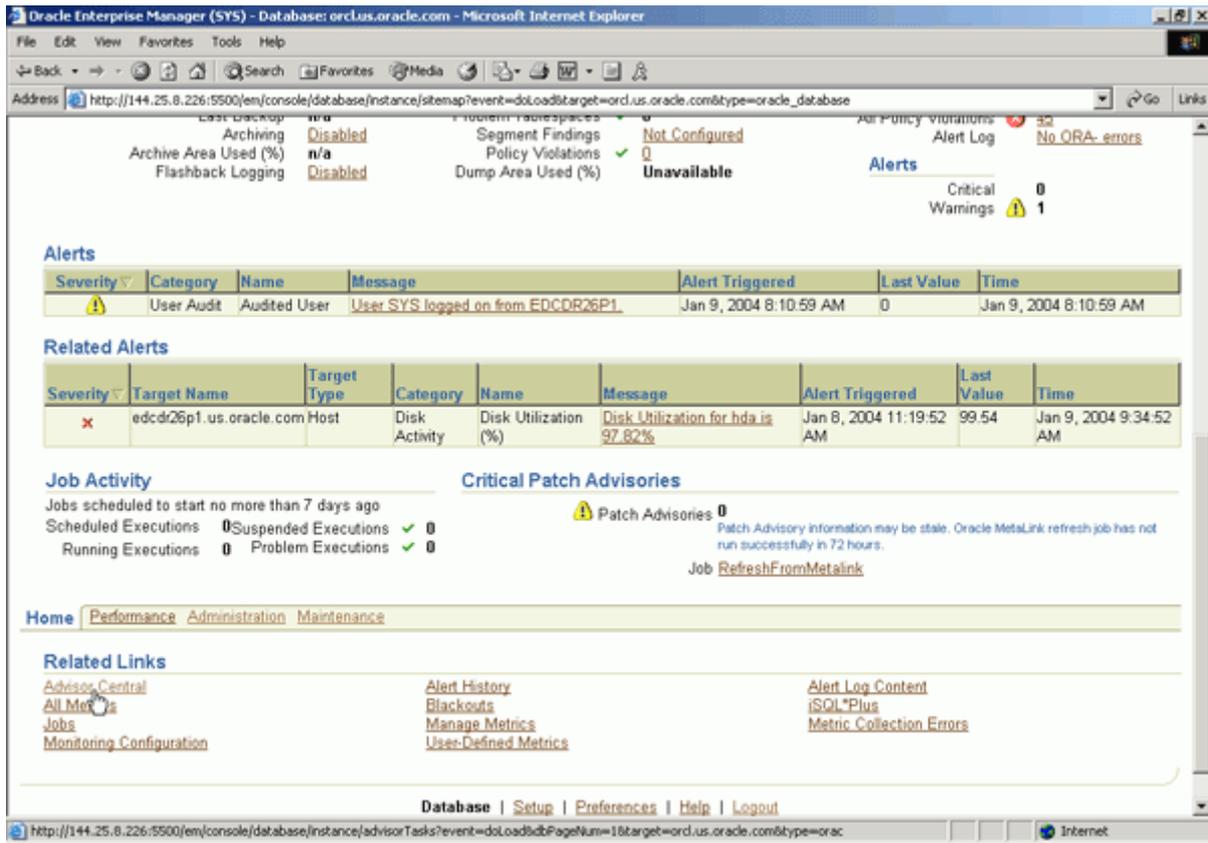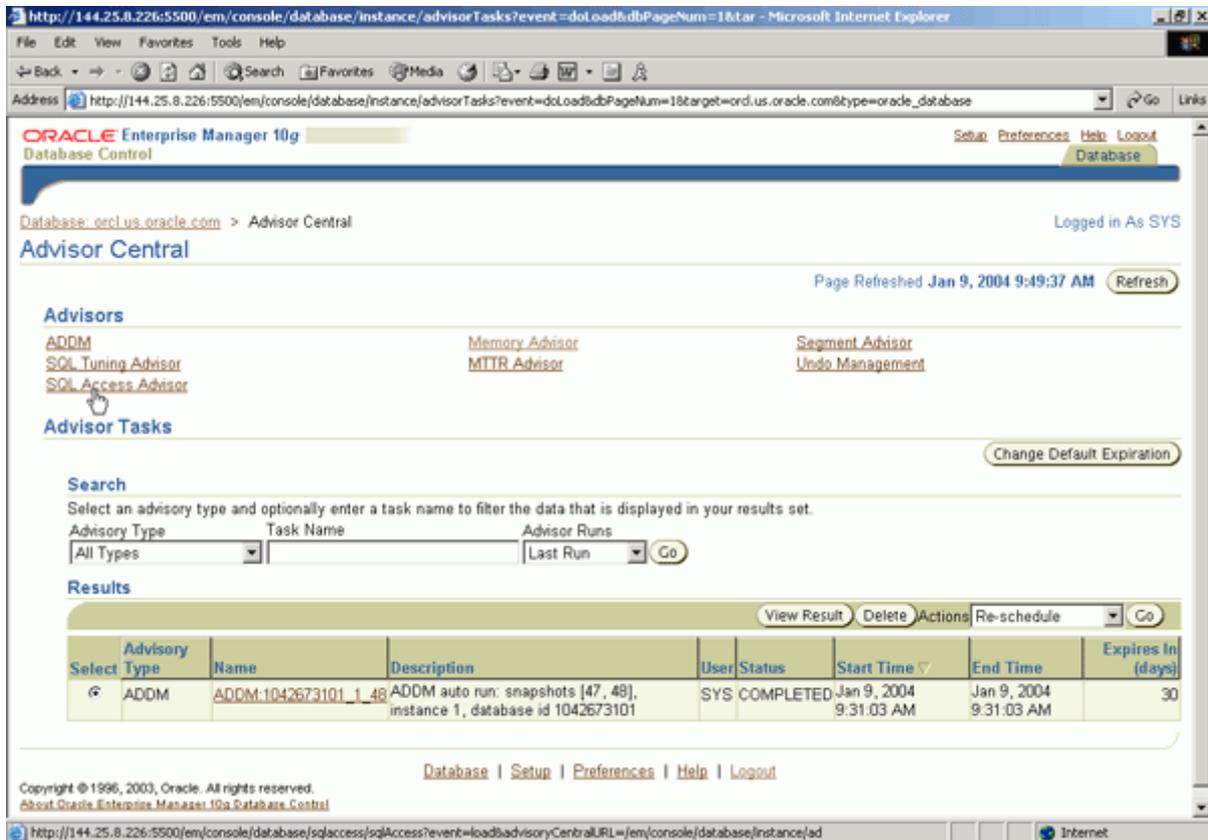
输入 sys/ 作为 SYSDBA ， 单击 Login 。
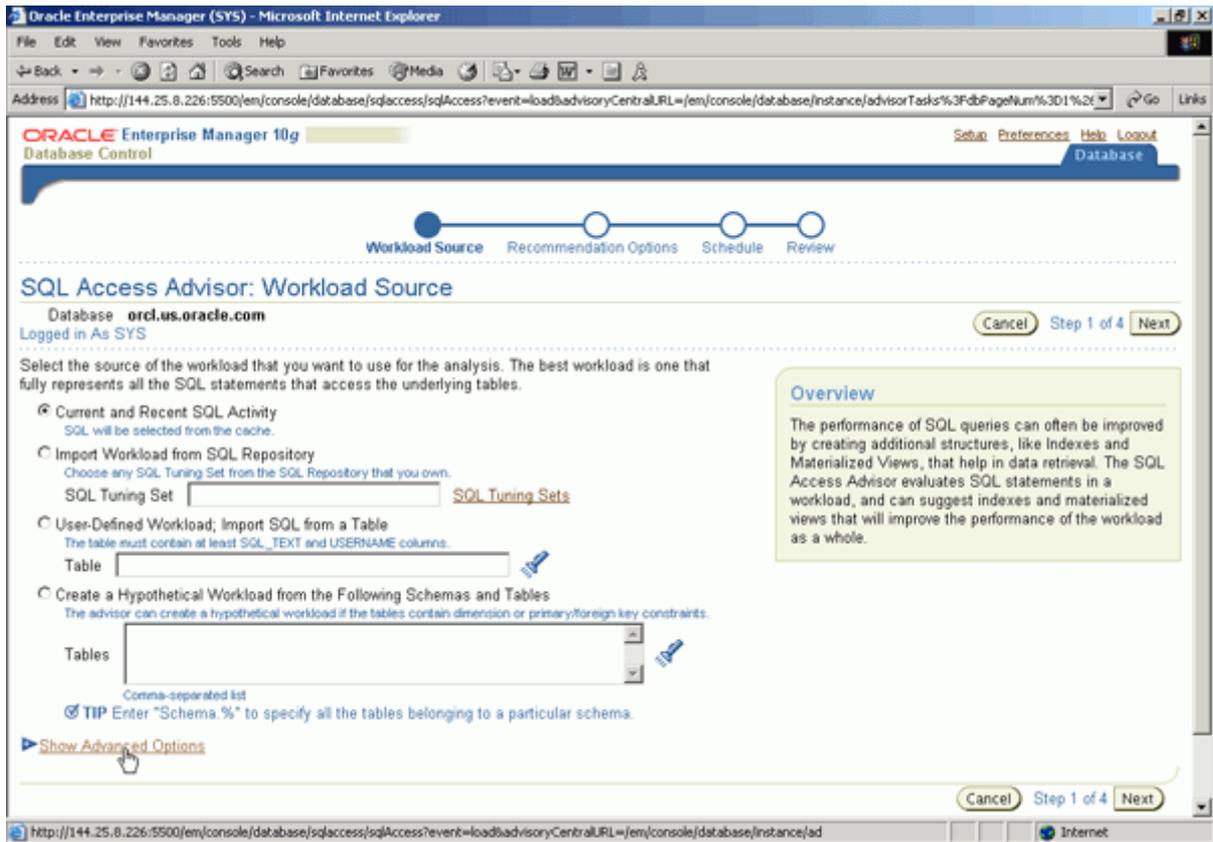
2. 向下滚动至主页底部，然后单击 Related Links 下的 Advisor Central 。
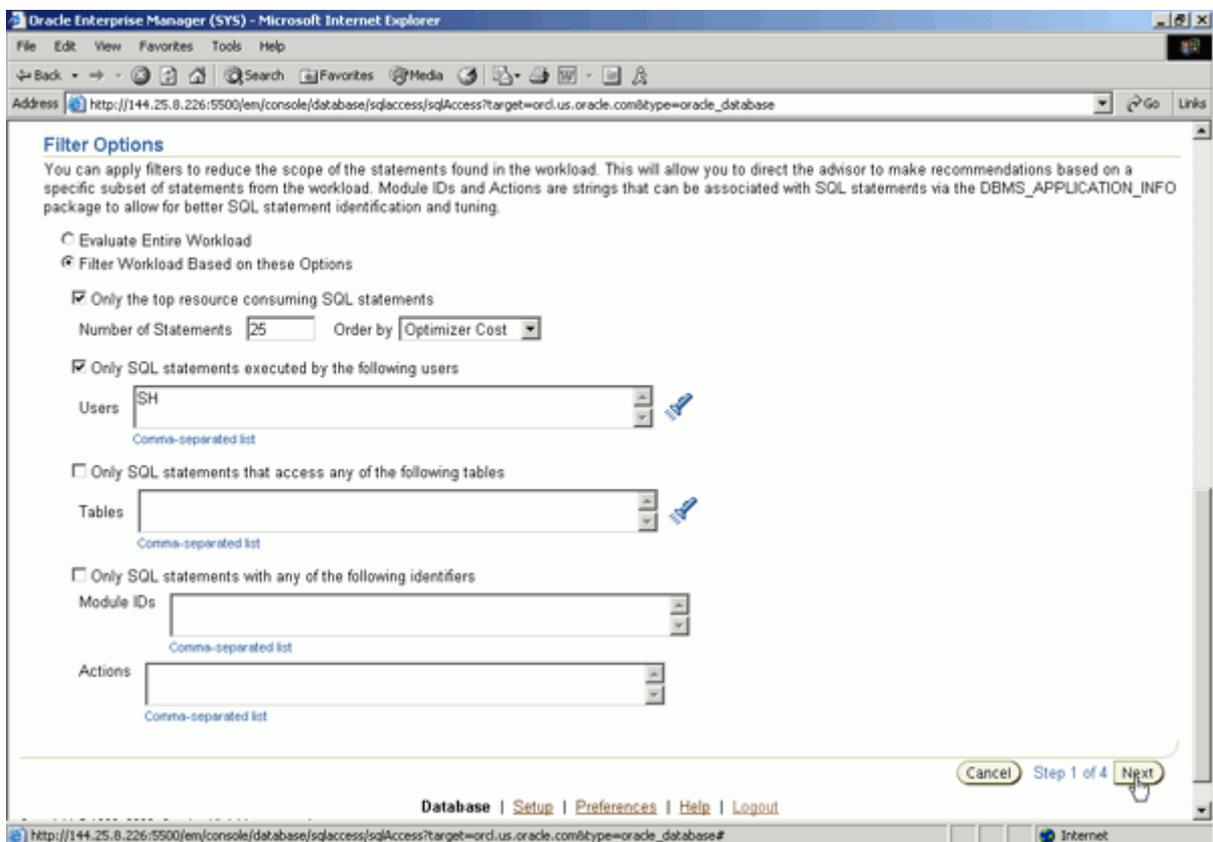
3. 单击 SQL Access Advisor 链接。
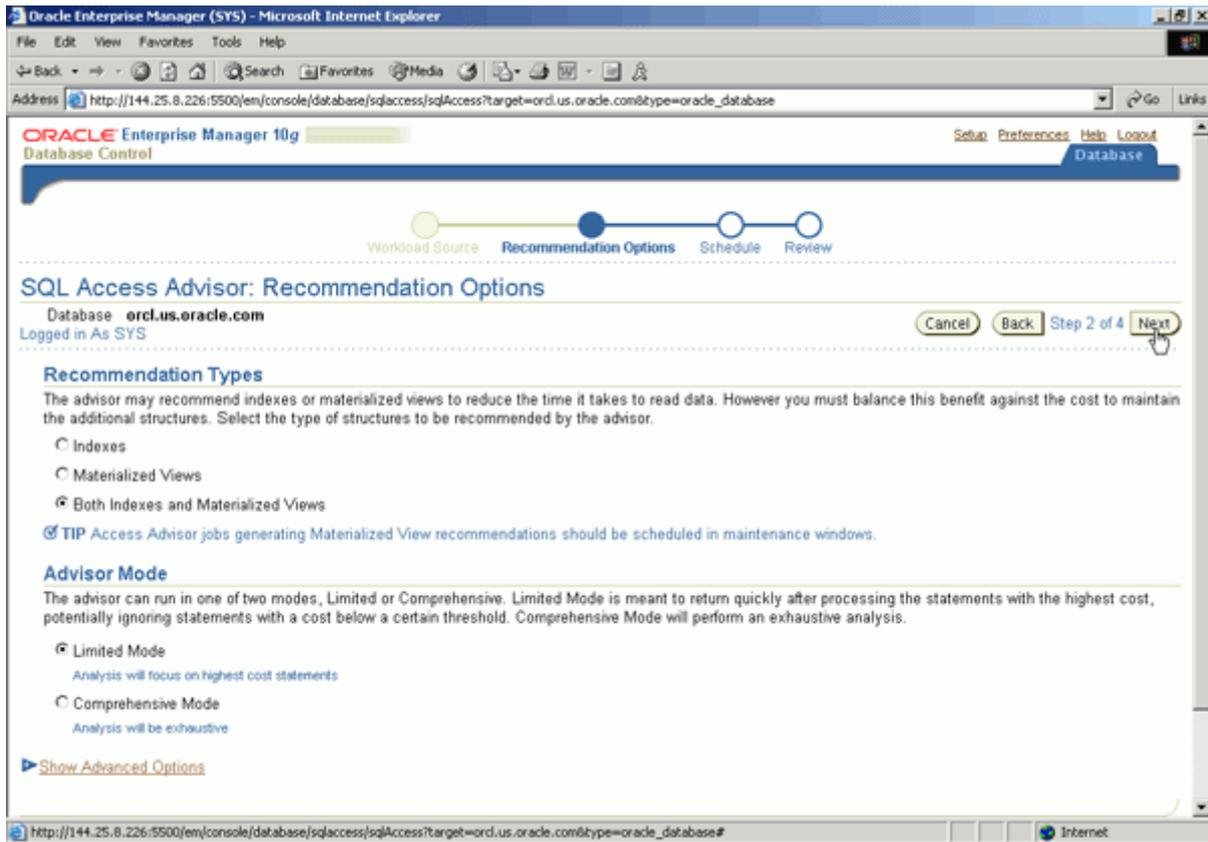
4. 确保选中了 Current and Recent SQL Activity ，然后单击 Show Advanced Options 。
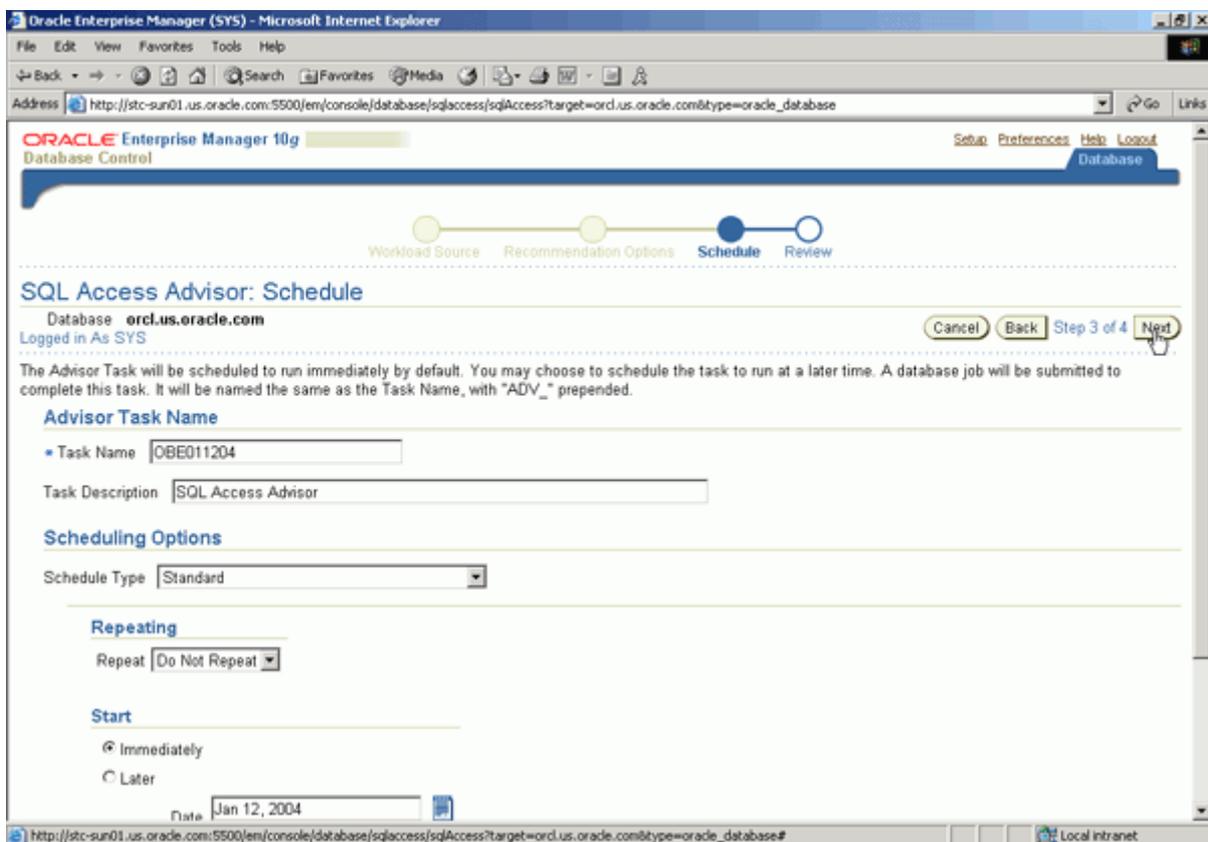
5. 向下滚动。 在 Filter Options 下，选择 Filter workload based on these options。 选择 Only the top resource consuming SQL statements 和 Only SQL statements executed by the following users 。 然后在 Users 域中输入 SH ，然后单击 Next 。
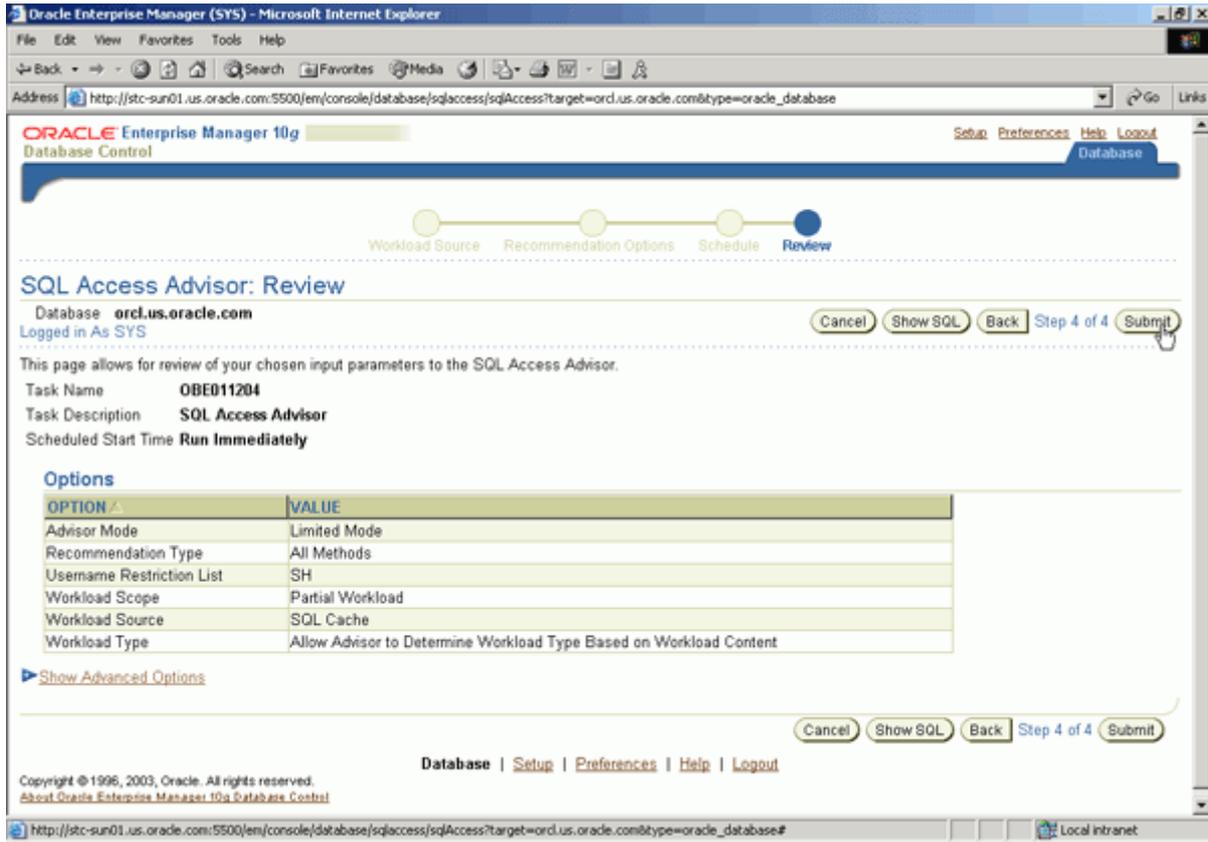
6. 选择 Both Indexes and Materialized Views ，然后单击 Next 。

7. 输入任务名 OBE ，为 Schedule Type 选择 Standard ，然后单击 Next 。
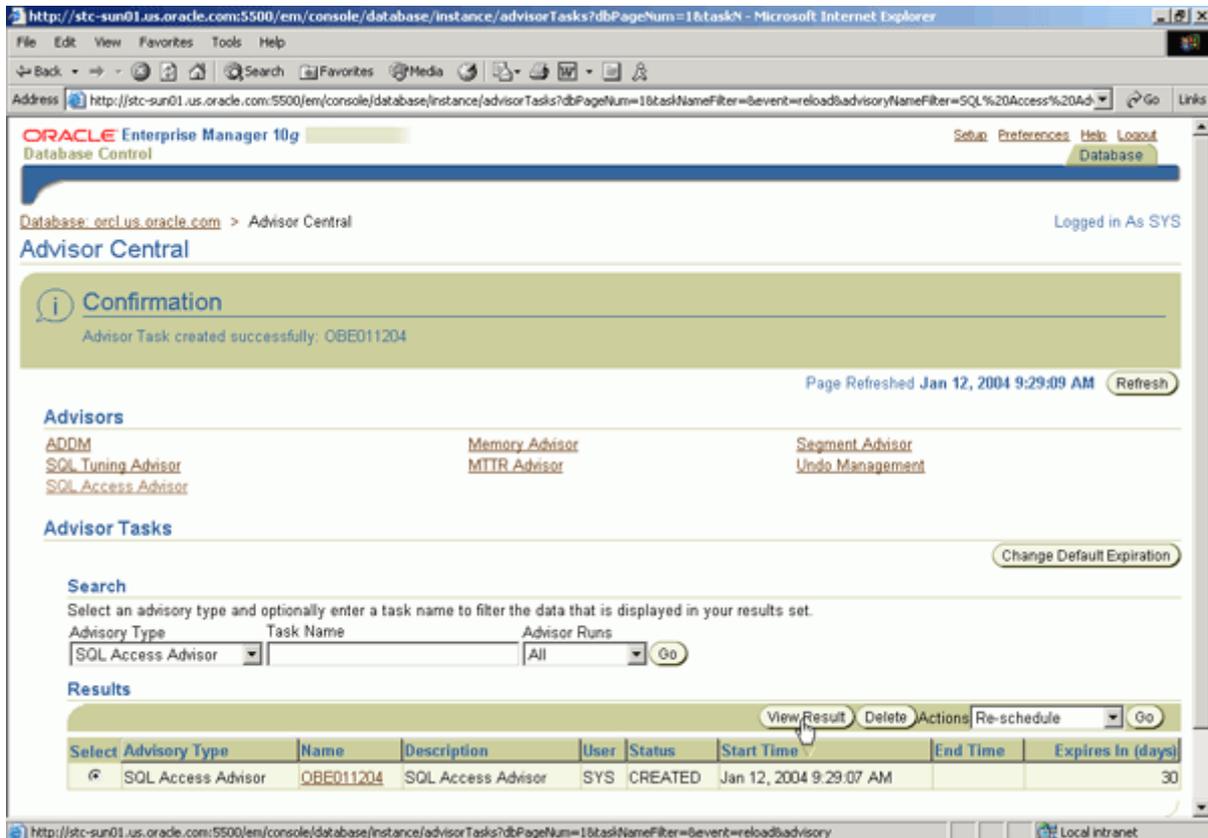
8. 在概要窗口中，单击 Submit 。
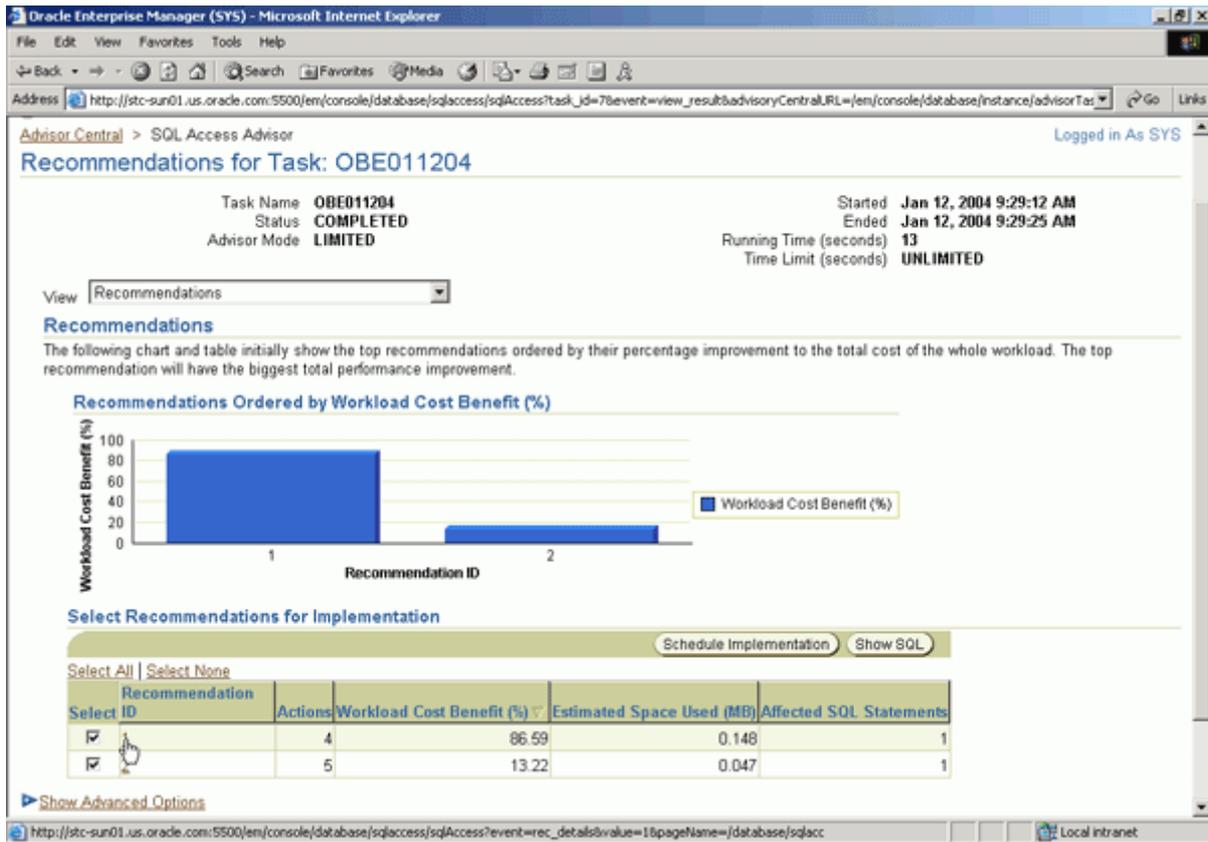
查看和执行建议

现在您可以查看结果，如果您希望,可以执行它们。 执行以下操作：
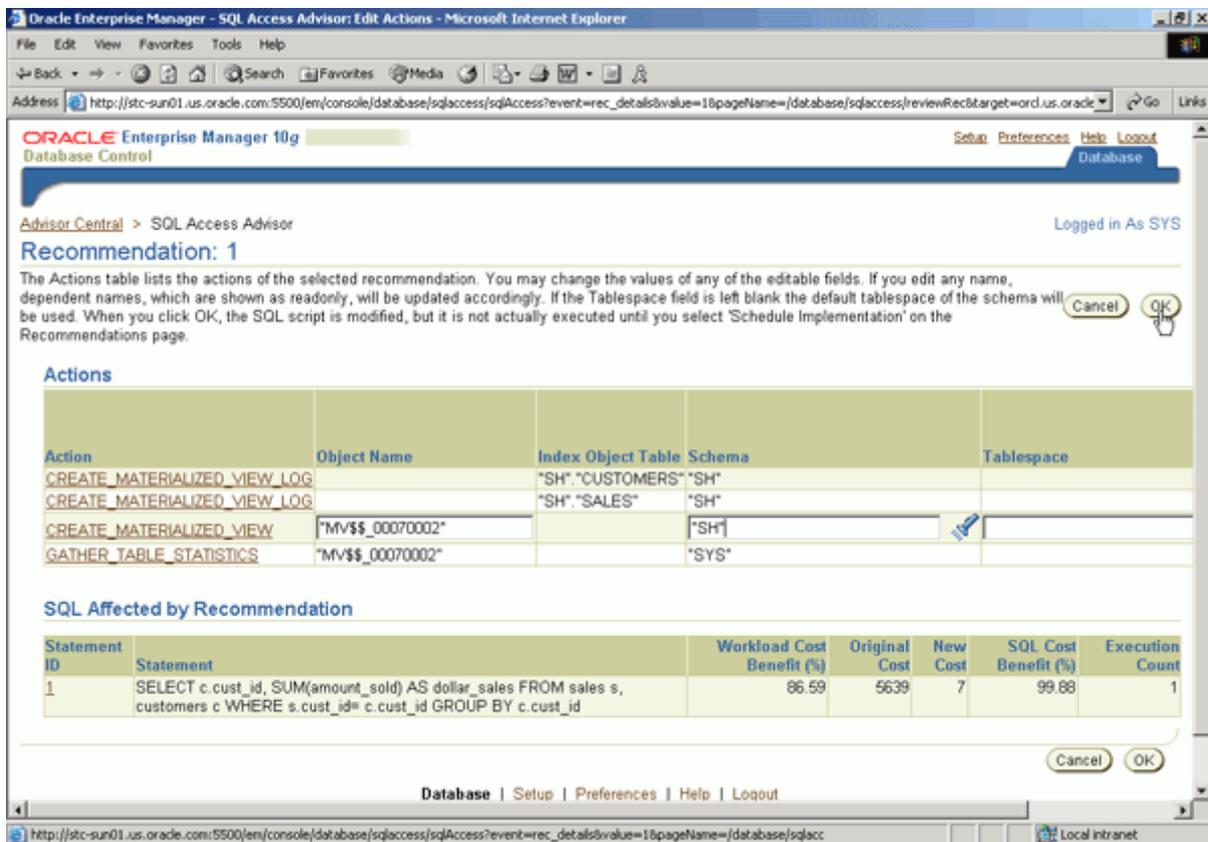
1. 确保选择了您的作业，然后单击 View Result 。
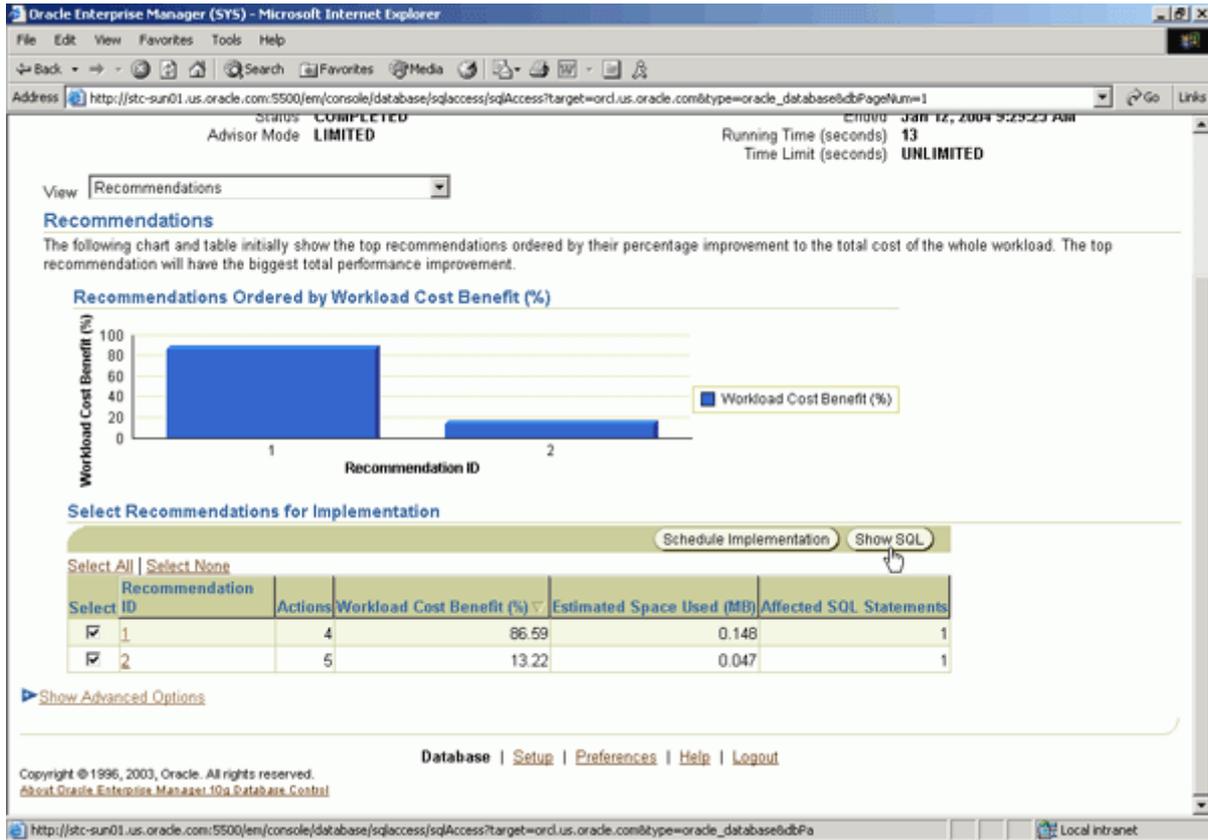
2. 单击 Recommendation ID 1 ，查看建议的细节。
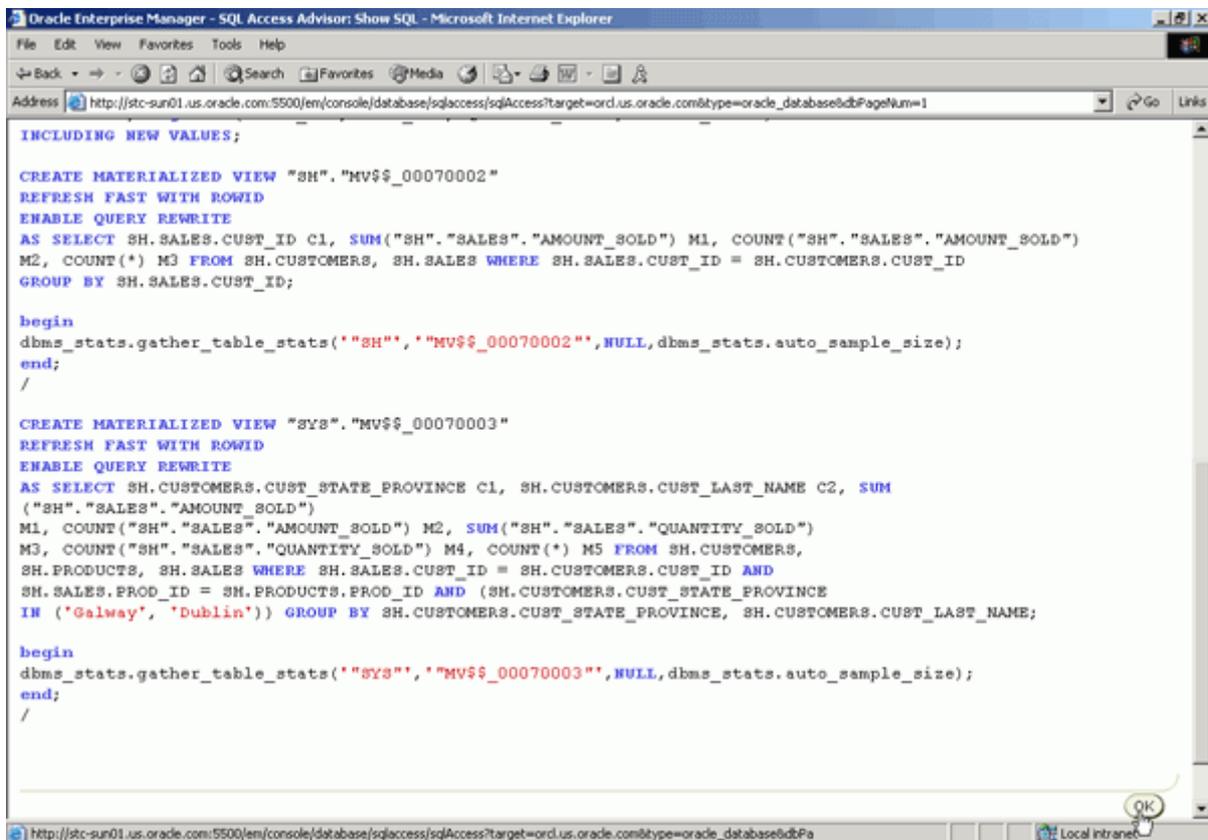
3. 在这里您可以自定义对象名称、模式和表空间，以执行建议。 向下滚动，并将 Create Materialized View 的 Schema Name 修改为 SH ，然后单击 OK 。
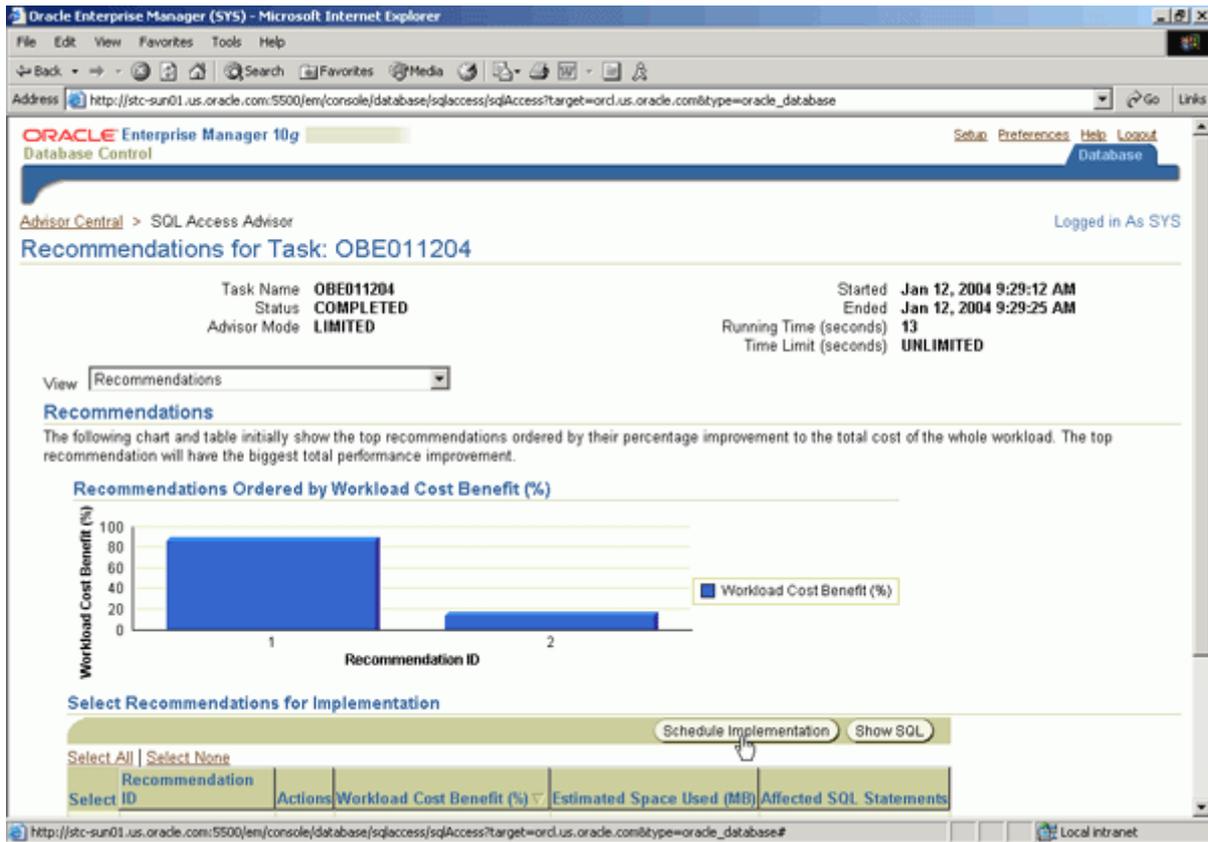
4. 当您安排执行时，如果要查看将执行的 SQL 脚本，可以单击 Show SQL 。
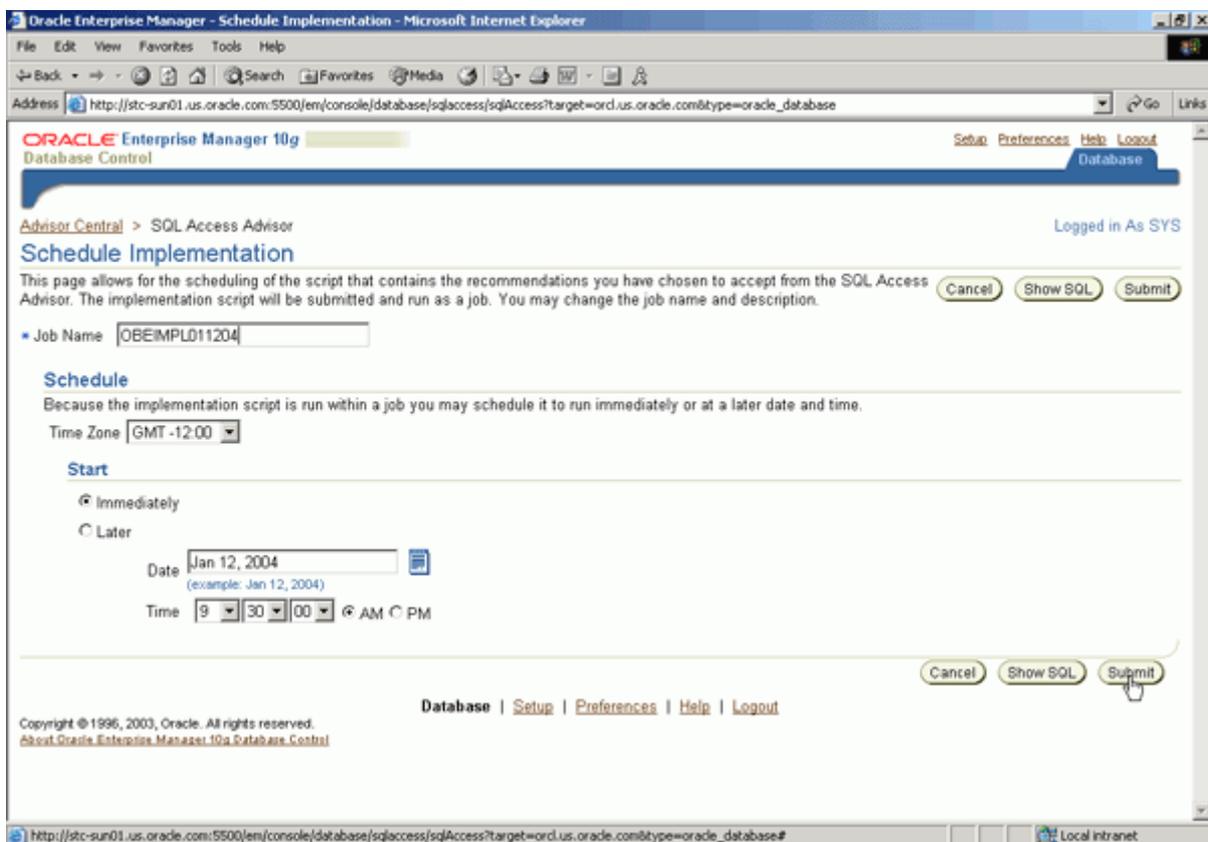
5. 向下滚动至底部，您将看到创建实体化视图的语句和您刚作的修改。单击 OK 。

6. 单击 Schedule Implementation ，执行建议。

7. 为 Job Name 输入 OBEIMPL ，然后单击 Submit 。

8. 执行作业创建完成并且现在正在运行。 单击 Run History 标签。

9. 确保选择了您的作业，然后单击 View 。

10. 查看概要，然后单击路径式导航栏中的 database 导航项 。
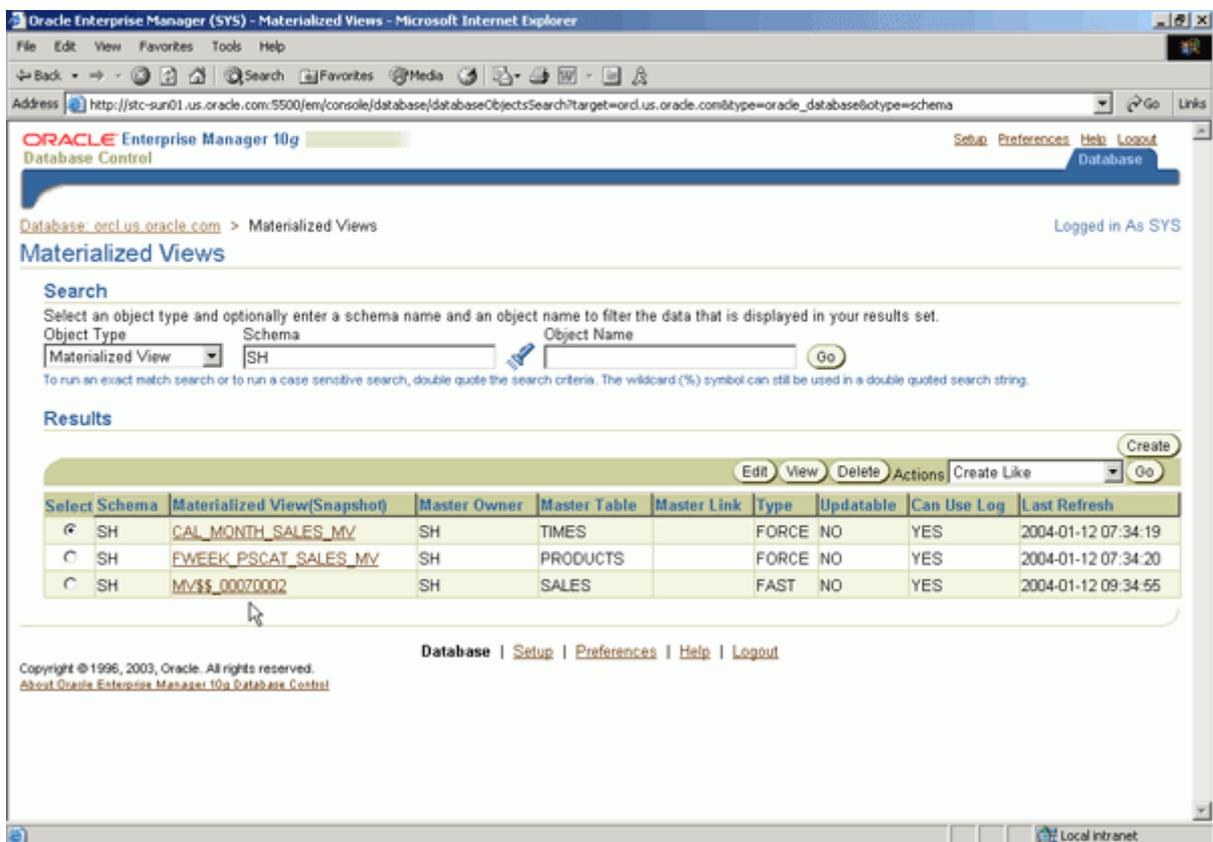
11. 单击 Materialized View 。

12. 在 Schema 域中输入 SH ，然后单击 Go 。

13. 注意新创建的实体化视图出现在列表中。单击路径式导航栏中的 Database 导航项,然后单击 Tables 链接。

使用 Memory Advisor

在这一部分中,您将主动管理并将一些与 Oracle 例程内存配置相关的任务自动化。 通过将内存配置自动化,您有了更多的时间来处理影响您的企业的应用或商务实际问题。
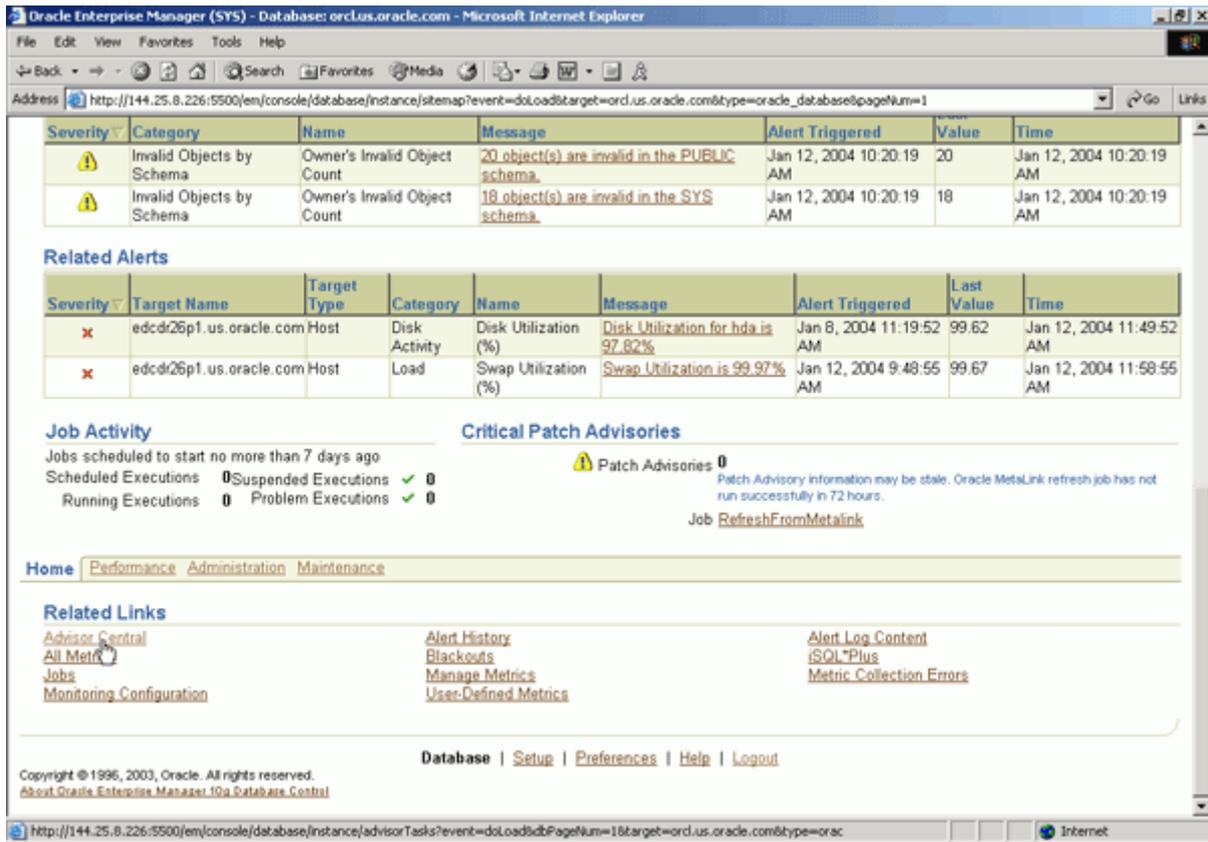
Memory Advisor 是 Oracle 数据库内部的一个智能专家系统，它为各种 SGA 和 PGA 组件主动确定最优设置。
当实现自动化时，Oracle 将根据工作负载的需求自动调整各种池和高速缓存的设置。

- 启用自动共享内容管理
- 修改总 SGA 大小
- 使用 PGA Advisor
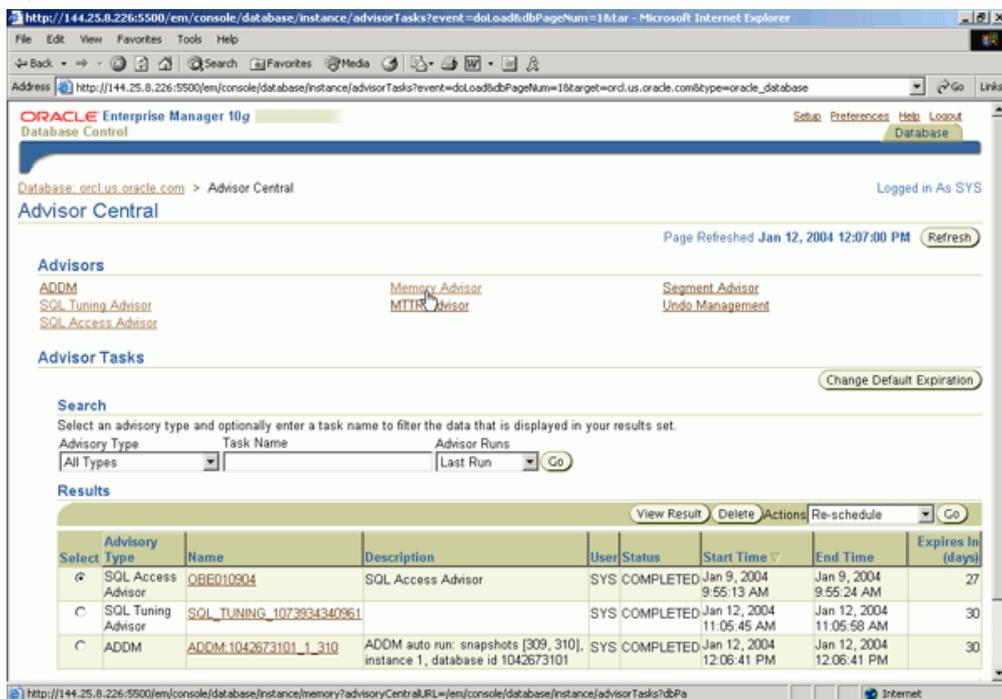
启用自动共享内存管理

执行以下操作，启用 SGA 的自动共享内存管理：

1. 向下滚动至主页底部，然后单击 Related Links 下的 Advisor Central 。

2. 选择 Memory Advisor 。

3. 为 Automatic Shared Memory Management 单击 Enable 。

4. 单击 OK ，启用自动共享内存管理。

5. Oracle 服务器现在将根据工作负载的需求自动调整各种池和高速缓存的设置。

修改总 SGA 大小

当处于自动共享内存管理模式下时，要修改总的 SGA 大小，您将需要确保最大的 SGA 大小足够大。 执行以下操作：

1. 向下滚动页面。 将 Maximum SGA Size 修改为 261 MB ，然后单击 Apply 。注意： 如果您接收到一个错误，那么单击 Refresh 按钮重试。

2. 单击 Yes ，确认修改。

修改总 SGA 大小

3. 提供主机证书和数据库证书。单击 OK 。

4. 当您修改 Max SGA Size 参数时，例程必须重新启动。 单击 Yes ，确认数据库重启。
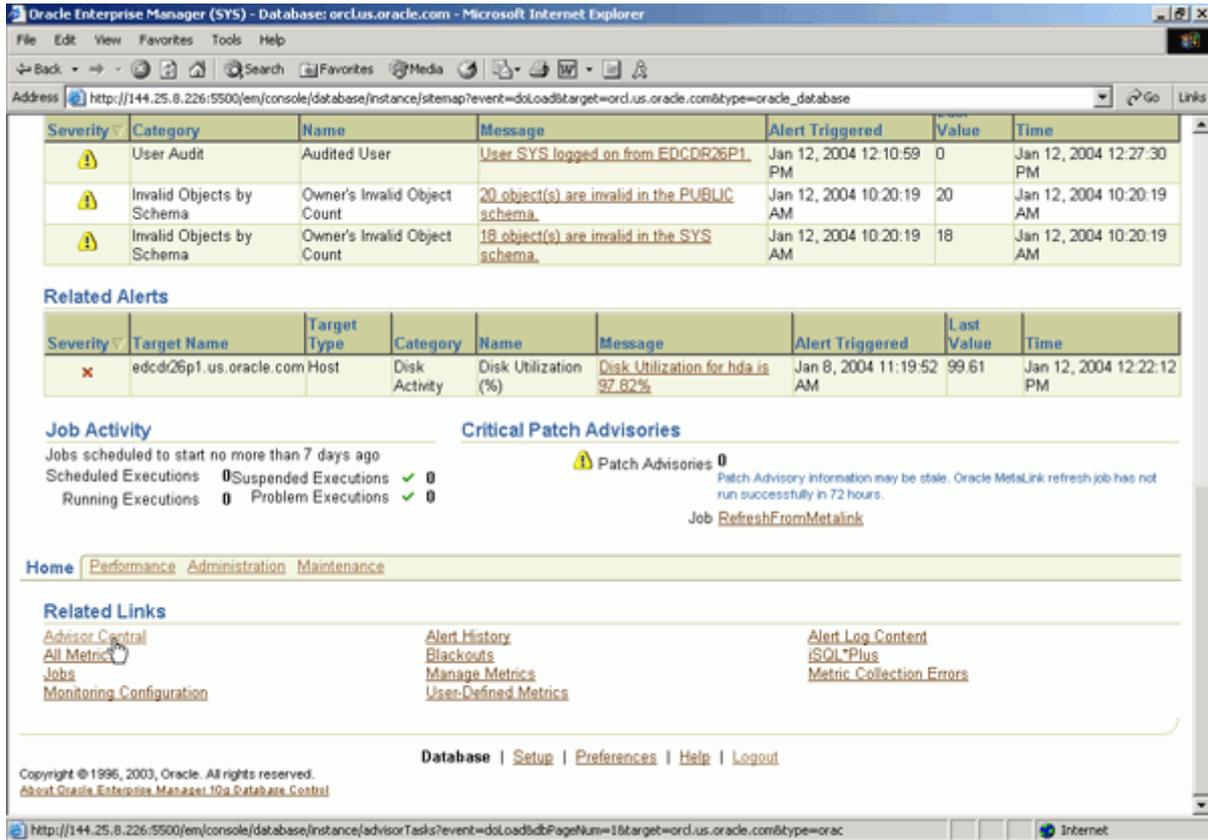
5. 数据库重启过程将开始。 单击 Refresh 。
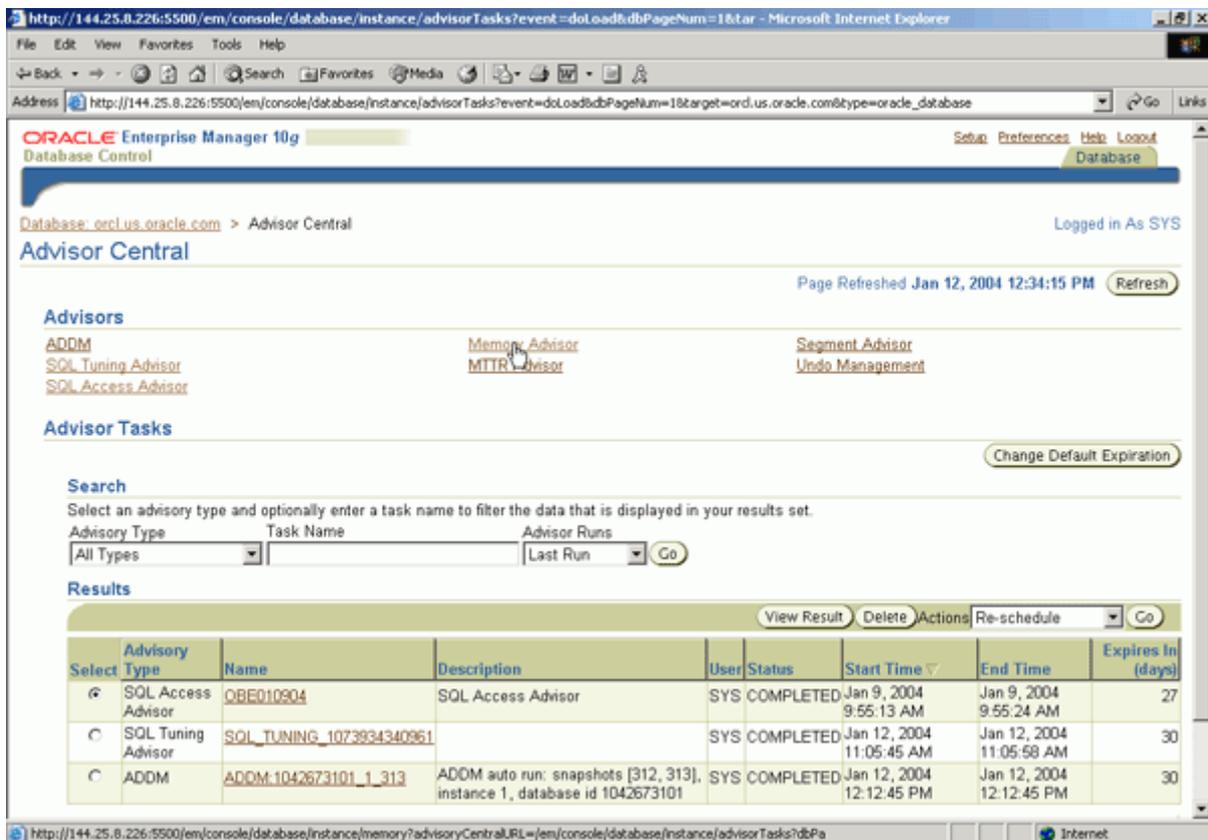
6. 输入 sys/ 作为 SYSDBA ，单击 Login 。

7. 向下滚动，然后单击 Advisor Central 。
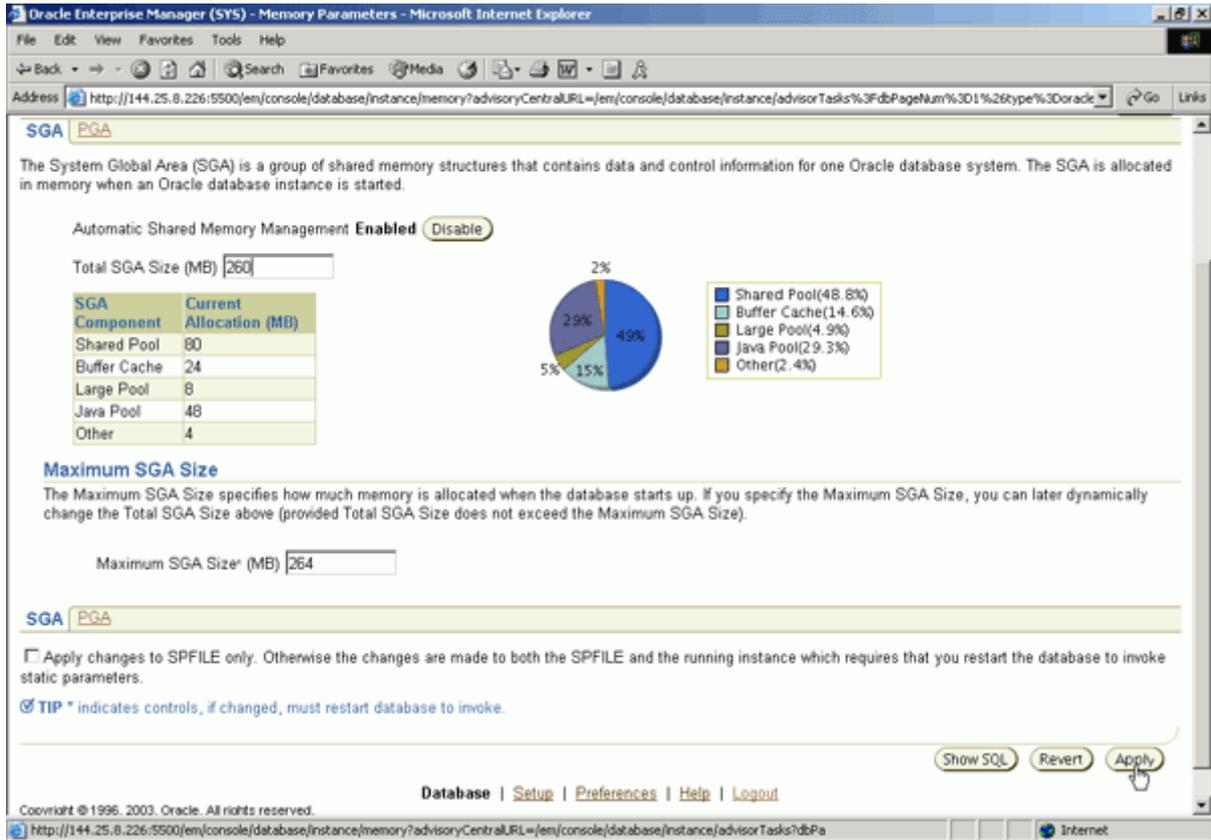
8. 单击 Memory Advisor 。
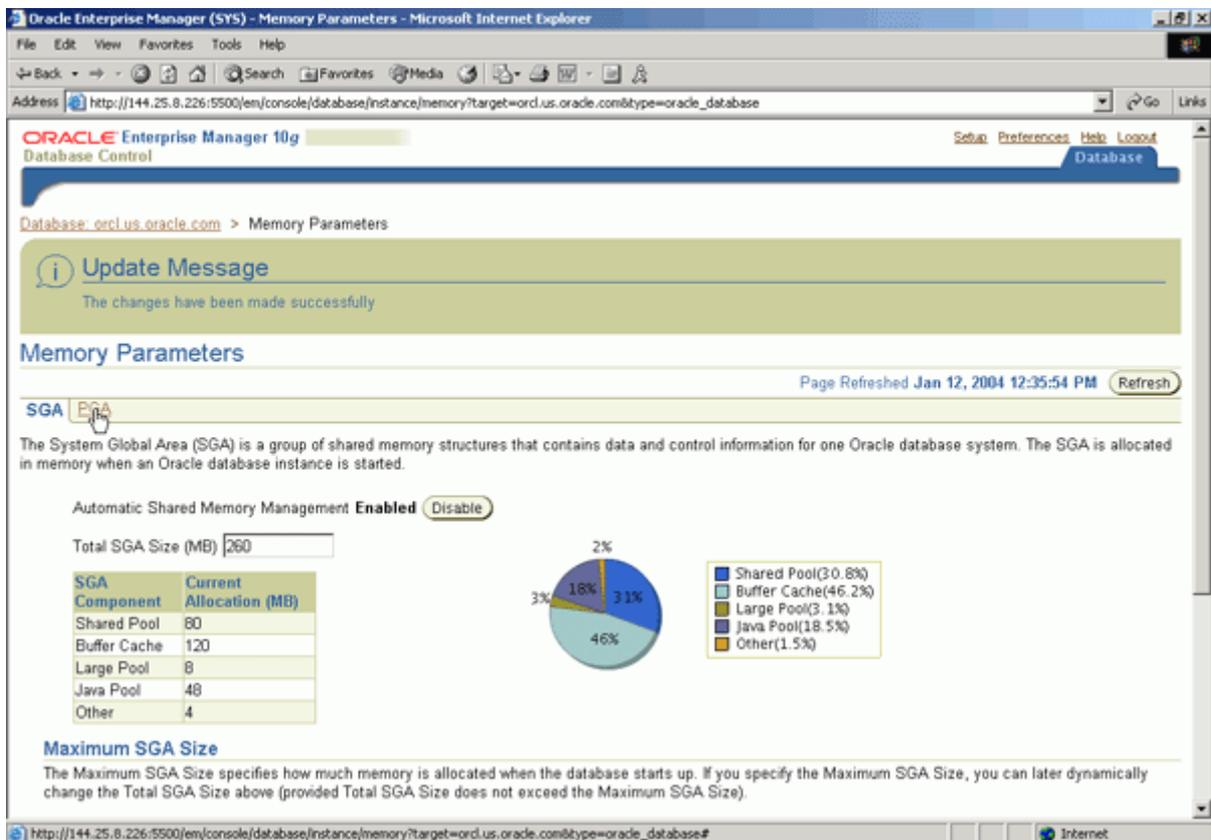
9. 将 Total SGA Size 参数修改为 260 MB。 单击 Apply 。

注意 Max SGA Size 参数自动调整与内存细粒大小保持一致，虽然您原来已把它设为 261 MB，但 Oracle 服务器自动将其修改为 264 MB。

10.　一旦您接收到了确认通告 — 指示参数已成功修改，您还将注意到一些 SGA 组件的内存分配已自动调整。

使用 PGA Advisor

　　执行以下操作，分配与 PGA 相关的内存：

1.　单击 PGA 。

2. 单击 Advice ……

3. PGA Aggregate Target Advice 图显示在高速缓存中找到数据（以便您不需访问磁盘）的频率。 在本文中，应当注意当前的 PGA Aggregate Size 被设为 24 MB 左右，所有请求的服务的 88% 以上都从内存中获得。 这还显示溢出范围从 12 MB 左右开始。 在 12 MB 时，PGA 请求应用高速缓存的概率大约为 90%。 PGA Aggregate Size 指示（根据当前的工作负载和数据库中的会话数量），在该数据库中为所有 PGA 分配的大小不超过 24 MB。单击 OK 。

4. 单击 PGA Memory Usage Details 。

5. 本图显示内存大小请求的使用率详情和各种 PGA 内存请求的执行百分率。 单击 OK 。

已完成