

Teamcenter Engineering Performance Tuning

Author	Paul Angier
Company	UGS
Email	paul.angier@ugs.com
Phone	651-482-2038

Premium Partners:



Microsoft

Agenda

- Scope of material
- Methodology
- Demonstration
 - Showing effect of incremental changes on MCAD manager performance
- Additional techniques
- Resources for field performance tuning
 - Documentation and tools

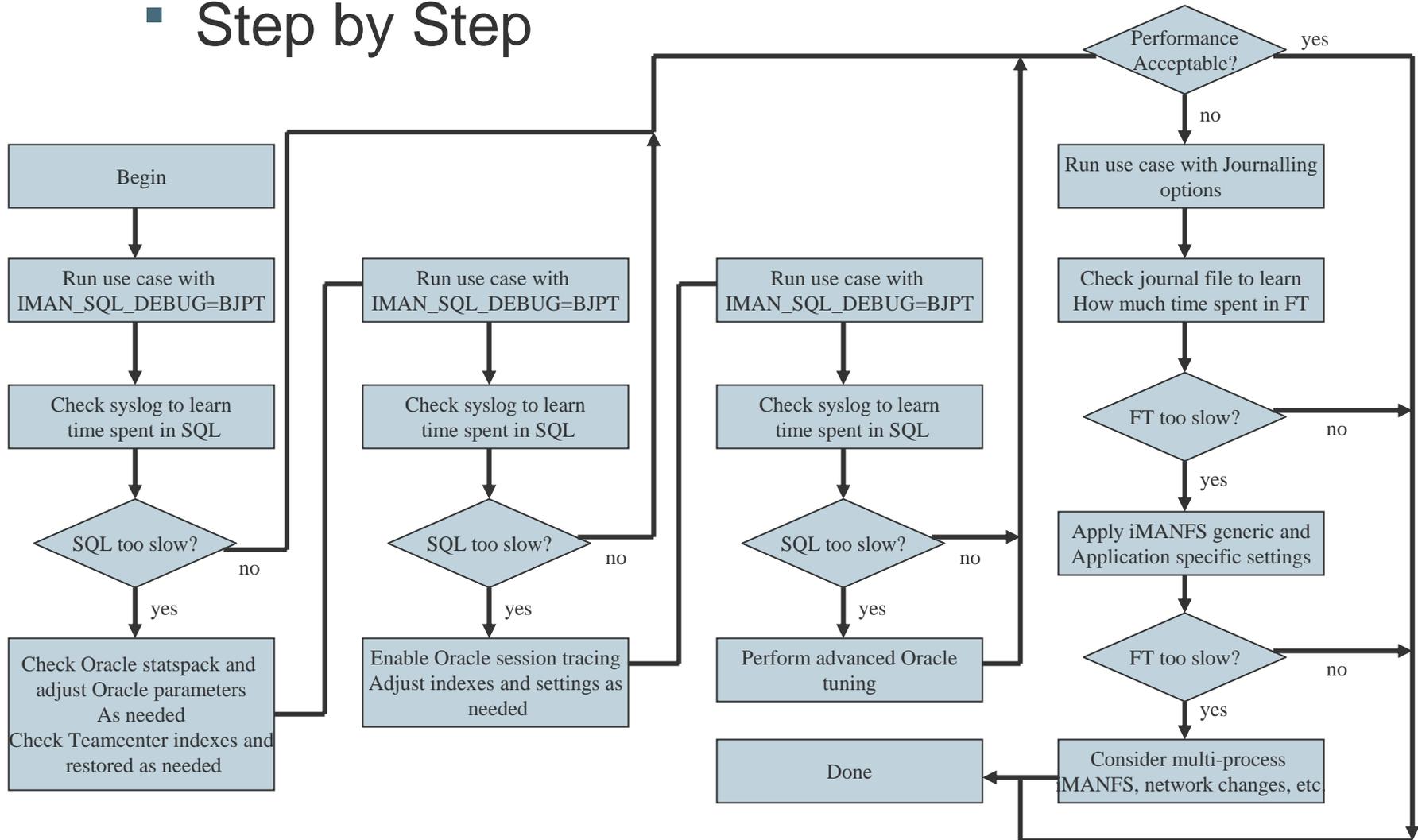
Scope

- V9.X and prior releases
- Single site only
- Field performance tuning
 - To reach a known baseline of performance tuning, prior to involving UGS support and/or product development
 - UGS Performance and Scalability team has advanced tools and techniques that are not covered here

- Goals of demonstration
 - Starting from a completely untuned system, show the effect of changing
 - Oracle server parameters
 - Teamcenter schema (indexes)
 - iMANFS configuration
 - Application-specific parameters
 - With emphasis on the systematic use of Teamcenter and Oracle logs to determine bottlenecks

Methodology

Step by Step



- All in one

- Enable syslog and journalling options
 - IMAN_SQL_DEBUG=Y
 - IMAN_SQL_DEBUG=BPT
 - IMAN_Journalling=ON
- Set TIMED_STATISTICS = TRUE in Oracle instance
- Execute beginning statspack snap
- Start TCEng session
- Activate Oracle session tracing
- Run test case, measure elapsed time
- Shut down TCEng session completely
- Execute ending statspack snap and generate the report
- Process session trace file with tkprof; be sure to specify “waits=yes” option

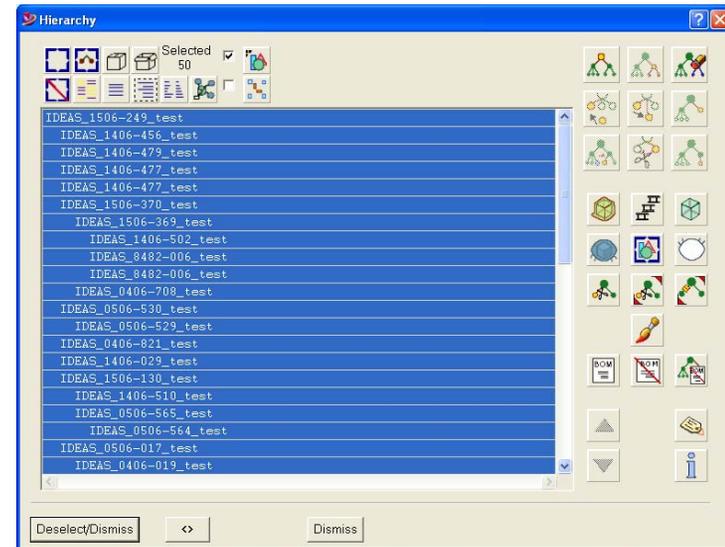
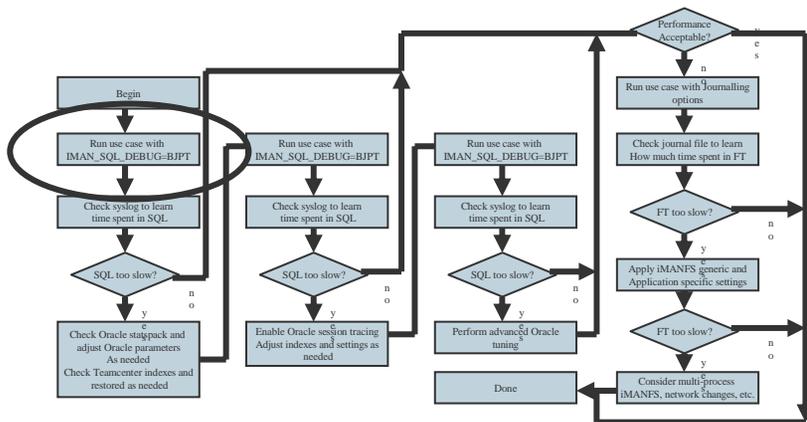
- Setting logging options - two types of imanserver logging
 - System log
 - On by default, but unless there is an error, the syslog is deleted when the session ends
 - Journalling
 - Off by default, must be enabled by configuration settings

- Where to set logging options
 - Logging options can be set in many places. It is easiest to set them in the %IPR%\server_configs file for the database you are connecting to
 - In a production system the %IPR% directory may be shared from a file server. In this case logging options should be set in the system control panel, as environment variables, on the test client
 - It is recommended to set logging options consistently in only one place, to avoid accidentally leaving them on and affecting performance.

- Capturing clean log files
 - For the purpose of analyzing a specific use case, it is essential to minimize the number of non-related actions
 - These will be reflected in the log file and will skew the results, making the traces more difficult to interpret
- Start the session
- Run the use case
- End the session, including the Orbix daemon
- Copy the logs to a safe location

Demonstration

- Open I-DEAS assembly from Teamcenter
 - Capture logs and elapsed time
 - Examine syslogs and statspack report to recognize
 - Oracle server bottlenecks
 - Missing OOTB indexes



Demonstration

- Teamcenter and Oracle log files
 - Teamcenter logs
 - Imanserver syslog
 - Oracle logs
 - Statspack report

STATSPACK report for

DB Name	DB Id	Instance	Inst Num	Release	Cluster	Host
a3p11	118899021	a3p11	1	9.2.0.6.0	NO	A30RAM3

	Snap Id	Snap Time	Sessions	Curs/Sess	Comment
Begin Snap:	22	23-Jun-05 09:48:34	209	6.7	
End Snap:	23	23-Jun-05 10:25:30	213	6.7	
Elapsed:		36.93 (mins)			

Cache Sizes (end)

Buffer cache:	664M	Std block size:	8K
shared pool size:	96M	Log buffer:	100K

Load Profile

	Per Second	Per Transaction
Redo size:	58,137.74	1,740.47
Logical reads:	7,964.14	238.42
Block changes:	427.47	12.80
Physical reads:	264.65	7.92
Physical writes:	21.04	0.63
User calls:	442.17	13.24
Parses:	406.70	12.18
Hard parses:	0.09	0.00
Sorts:	30.33	0.91
Logons:	0.01	0.00
Executes:	407.73	12.21
Transactions:	33.40	

% blocks changed per read:	5.37	Recursive call %:	8.60
Rollback per transaction %:	0.16	Rows per sort:	5.22

Instance Efficiency Percentages (Target 100%)

buffer nowait %:	100.00	redo nowait %:	100.00
buffer hit %:	96.68	in-memory sort %:	100.00
library hit %:	99.99	soft parse %:	99.98
execute to parse %:	0.25	latch hit %:	99.86
parse CPU to parse elapsed %:	94.28	% non-parse CPU:	99.92

```
SELECT vuid FROM POKL_LOCK MINUS SELECT vuid FROM POKL_LOCK WHERE type=
2 ORDER BY vuid ;
MFR40 POWS4 0.000 SECS;
DOC_exit (860 Insts, 11 Ghosts)
SM Exit (202 Opns, 0 Sops)
DELETE FROM PM_PROCESS_LIST WHERE pm_process_id = 'AABJUTJ3VAZDA';
MFR40 POWS4 0.000 SECS;
```

Line	Freq	Cur.t	Max.t	Avg.t	Rows	Err.	Statement
1	10	0.015	0.021	0.002	0	0	SELECT DISTINCT t_0.vuid FROM PMINSTANCE t_0 WHERE (UPPER(t_0.psysp_name) = UPPER(i1))
2	6	0.000	0.000	0.000	0	0	SELECT poid, puid FROM PPM_OBJECT WHERE puid IN (i1)
3	4	0.011	0.011	0.004	4	0	SELECT t_01.vuid FROM PPM_APPLICATION t_01 WHERE ((t_01.pname = i1) AND (t_01.ppassword = i2))
4	3	0.000	0.000	0.000	3	0	SELECT puid,ptimestamp FROM PPM_OBJECT WHERE puid IN (i1) FOR UPDATE
5	3	0.000	0.000	0.000	3	0	SELECT DISTINCT t_0.vuid FROM PPM_MEMBER t_0 WHERE ((t_0.ruseru = i3) AND (t_0.rgroupu = i2))
6	3	0.000	0.000	0.000	3	0	INSERT INTO POKL_LOCK (ref_vuid, ref_class, vuid, process, type) VALUES (i1,i2,i3,i4,i5)
7	3	0.000	0.000	0.000	3	0	DELETE FROM POKL_LOCK WHERE process = i1 AND ref_vuid = i2
8	3	0.000	0.000	0.000	2	0	COMMIT
9	2	0.000	0.000	0.000	0	0	SELECT t_01.vuid FROM PPM_APPLICATION t_01 WHERE t_01.pname LIKE i1
10	2	0.000	0.000	0.000	0	0	SELECT t_02.vuid FROM PMINSTANCE t_02 WHERE (t_02.pname = i1)
11	2	0.000	0.000	0.000	0	0	SELECT t_01.vuid FROM PPM_PACKAGE t_01 WHERE (t_01.PACK_NAME = i1)
12	2	0.000	0.000	0.000	30	0	SELECT vuid FROM PPM_PRIVILEGES
13	2	0.000	0.000	0.000	0	0	SELECT poid FROM PPM_CLASS WHERE UPPER(PMCLASS.pname) = UPPER('linfofonage')
14	2	0.000	0.000	0.000	2	0	SELECT PLIST_OF_ROLE.puid,PRIVILEGE.VALID,PLIST_OF_ROLE.PTIMESTAMP FROM EXIST_OF_ROLE = PPM_OBJECT WHERE
15	2	0.000	0.000	0.000	0	0	PLIST_OF_ROLE.puid = PPM_OBJECT.puid AND PLIST_OF_ROLE.puid IN (i1)
16	2	0.000	0.000	0.000	0	0	SELECT PPM_MEMBER, puid, ptimestamp, puid, ptimestamp, puid, rowing_sitcu, rowing_sitcu, rgroupu, rgroupu,
17	2	0.000	0.000	0.000	0	0	PPM_MEMBER, puid, rgroupu, rgroupu, puid, ptimestamp, puid, ptimestamp, puid, ptimestamp, puid, ptimestamp, puid, ptimestamp,
18	2	0.000	0.000	0.000	0	0	SELECT DISTINCT t_01.vuid FROM PPMINSTANCE t_01 WHERE (UPPER(t_01.psysp_name) = UPPER(i1))
19	2	0.000	0.000	0.000	0	0	SELECT DISTINCT t_02.rroulu FROM PPM_MEMBER t_02, PPM_MEMBER t_02 WHERE (((t_02.rgroupu = i1) AND
20	1	0.000	0.000	0.000	0	0	(((t_02.ruseru = i2) AND (t_02.pstatus = i3)) AND t_02.rroulu IS NOT NULL) AND (t_02.rroulu = i4)))
21	1	0.000	0.000	0.000	0	0	SELECT PPM_ARGUMENTS.package_name, p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, p13, p14, p15, p16, p17, p18, p19, p20,
22	1	0.000	0.000	0.000	0	0	SELECT PPM_ARGUMENTS.package_name, p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, p13, p14, p15, p16, p17, p18, p19, p20,
23	1	0.000	0.000	0.000	0	0	SELECT puid FROM PPMINSTANCE WHERE NOT EXISTS (SELECT t_01.vuid FROM PPMINSTANCE WHERE t_01.vuid = puid)
24	1	0.000	0.000	0.000	0	0	SELECT puid FROM PPMINSTANCE WHERE NOT EXISTS (SELECT t_01.vuid FROM PPMINSTANCE WHERE t_01.vuid = puid)
25	1	0.000	0.000	0.000	0	0	SELECT puid FROM PPMINSTANCE WHERE NOT EXISTS (SELECT t_01.vuid FROM PPMINSTANCE WHERE t_01.vuid = puid)
26	1	0.000	0.000	0.000	0	0	SELECT puid FROM PPMINSTANCE WHERE NOT EXISTS (SELECT t_01.vuid FROM PPMINSTANCE WHERE t_01.vuid = puid)
27	1	0.015	0.015	0.015	1	0	SELECT pname FROM PPM_CLASS WHERE UPPER(pname) = 'CPMINTENTENTRY'
28	1	0.000	0.000	0.000	1	0	SELECT pname FROM PPM_CLASS WHERE UPPER(pname) = 'CPMINTENTENTRY'
29	1	0.000	0.000	0.000	1	0	SELECT pname FROM PPM_CLASS WHERE UPPER(pname) = 'CPMVERIDENTRY'
30	1	0.000	0.000	0.000	1	0	SELECT pname FROM PPM_CLASS WHERE UPPER(pname) = 'CPMATESTENTRY'

Demonstration

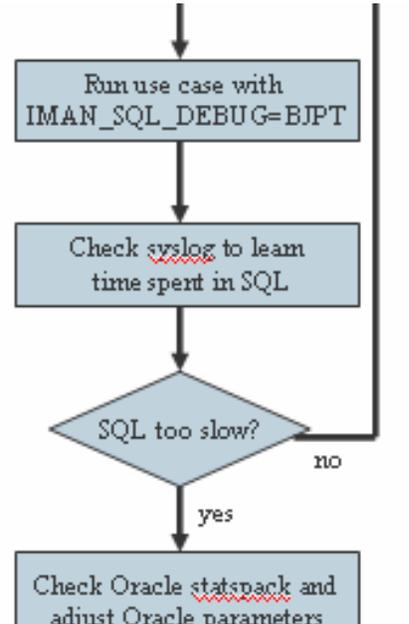
- Imanserver syslog
 - What does it tell you?
 - Oracle server initialization parameters
 - Teamcenter environment variable settings
 - SQL details
 - What queries were run during the session
 - Detailed statistics for each query
 - How much time was spent in SQL
 - Optional trace information such as AM rules evaluation (not directly related to performance tuning)
 - All information reported from the imanserver point of view

Demonstration

- Oracle statspack report
 - What does it tell you?
 - Details about performance of Oracle server over the elapsed time period
 - Buffer cache performance
 - Most time-consuming events
 - Query performance
 - Much much more
 - Important to realize that the statspack report is not specific to any one Teamcenter session or use case, unless you are in a single-user environment
 - Everything is from the Oracle server point of view – statspack knows nothing about Teamcenter

Demonstration

- How do you know whether SQL performance is too slow?



Demonstration

- Calculate what the underlying network will support, independent of Teamcenter and Oracle overhead
 - For example, if ping and/or tracert shows the latency between imanserver and Oracle is 91 ms on average . .

This is an extreme example - we would never recommend attempting to run a two-tier client over this link!

```
cmd
Minimum = 85ms, Maximum = 103ms, Average = 94ms
C:\Temp>ping moshp1
Pinging moshp1.sdrc.com [146.122.70.27] with 32 bytes of data:
Reply from 146.122.70.27: bytes=32 time=89ms TTL=245
Reply from 146.122.70.27: bytes=32 time=74ms TTL=245
Reply from 146.122.70.27: bytes=32 time=107ms TTL=245
Reply from 146.122.70.27: bytes=32 time=97ms TTL=245
Ping statistics for 146.122.70.27:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 74ms, Maximum = 107ms, Average = 91ms
C:\Temp>
```

Demonstration

- And the syslog shows that we are making 3304 calls to Oracle, for a particular use case

This is pretty good SQL performance, considering the latency

```
imanserver.exe56a803a0.syslog - Notepad
File Edit Format View Help
rowning_sitec,pis_frozen) VALUES (:1,:2,:3,:4,:5,:6)
000 1 0.078 0.078 0.078 1 0 INSERT INTO POM_M_LOCK (ref_uid, ref_class, vuid, process, type)
:4,:5)
94 0.094 0.094 1 0 INSERT INTO PM_PROCESS_LIST (pm_process_id, pm_key, pm_module)
:3)
52 0.062 0.062 1 0 DELETE FROM PPOM_SESSION WHERE puid = :1
78 0.078 0.078 1 0 DELETE FROM PPOM_OBJECT WHERE puid = :1
78 0.078 0.078 5 0 DELETE FROM POM_R_LOCK WHERE process = :1
79 0.079 0.079 1 0 DELETE FROM POM_M_LOCK WHERE process = :1 AND ref_uid = :2
78 0.078 0.078 0 0 DELETE FROM POM_M_LOCK WHERE process = :1
78 0.078 0.078 0 0 DELETE FROM POM_B_LOCK WHERE process = :1
78 0.078 0.078 0 0 DELETE FROM POM_BACKPOINTER WHERE from_uid = :1
219 0.094 0.094 1 0 DELETE FROM PM_PROCESS_LIST WHERE pm_process_id = :1
220 1.189 1.189 0 0 CONNECT
220 0.078 0.078 1 0 BEGIN eim_uids.new_fid(:uid); END;

Sum: 3304 312.253 7433 0

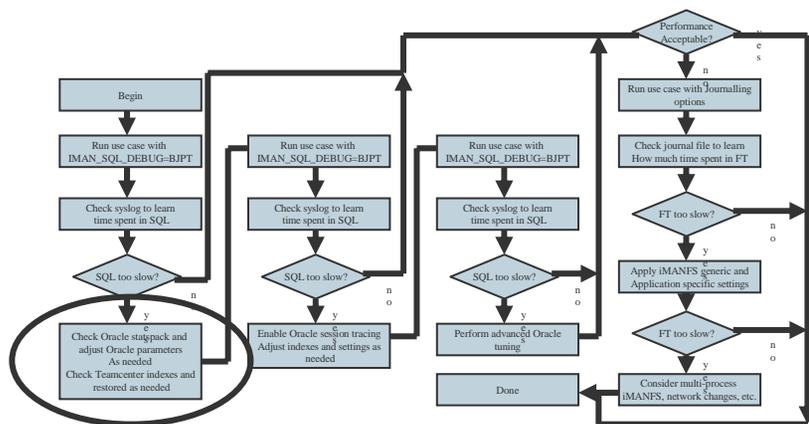
COMMIT;
err=0 rows=1 0.078 secs;
**** (Commit) DB Transaction Time was 1.453000 sec (cpu 0.000000)
2 seconds to POM DB x-act
@@@ End of session Fri Apr 14 08:42:33 2006 Central Daylight Time
```

- Then we cannot expect the SQL part of the use case to run any faster than 300 seconds

$$.091 * 3304 = 300.664$$

Demonstration

- Change Oracle initialization parameters
- Change SQLNet parameters
- Restore missing OOTB indexes
- Run analyze to rebuild Oracle statistics



- Oracle initialization parameters
 - Set to default values during installation, if Teamcenter templates are used
 - Selectively change values based on site characteristics
 - Set to default Oracle values or user-specified values if Teamcenter templates are NOT used
 - Resulting values usually will not be appropriate for Teamcenter – must validate and change if necessary
 - After changing certain Oracle parameters, or making index changes, you must rebuild statistics. These are needed by the CBO (cost based optimizer) to execute queries efficiently. Statistics must be rebuilt periodically to account for changing table sizes, data skewing, etc.

Demonstration

- SQLnet parameters
 - Oracle uses default packet sizes that are unnecessarily small for reliable networks.
 - Changed by setting SDU and TDU in listener.ora on Oracler server, and in tnsnames.ora in \$IMAN_DATA
 - Default is 2048 bytes
 - Can calculate optimal value, but standard recommendation is 8192 bytes
 - Oracle will wait until buffer is full before sending requests
 - Use tcp.nodelay to force immediate send

Demonstration

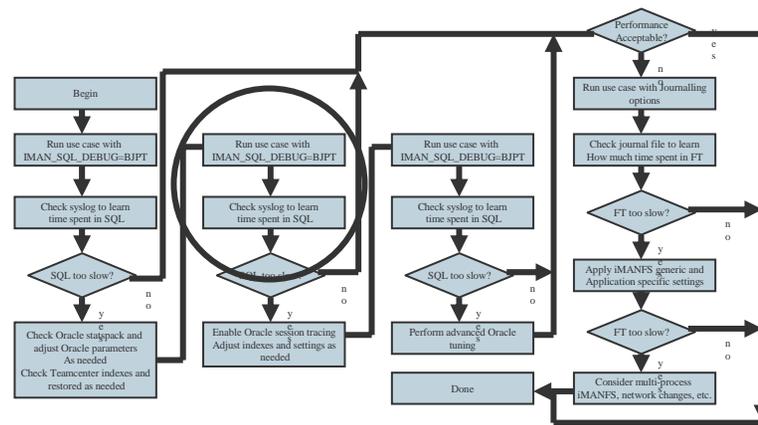
- Restore any missing OOTB indexes
 - Missing indexes are common at sites that have upgraded from one version of Teamcenter to another
 - Index_verifier utility must be used
 - Reports missing indexes by comparing database schema with OOTB definitions
 - Generates SQL statements to restore missing indexes
 - Index_verifier can be run at any time. By default it only reports, it does not make any changes. There is no effect on users.

Demonstration

- Rebuilding Oracle statistics
 - After changing certain Oracle parameters, or making index changes, you must rebuild statistics. These are needed by the CBO (cost based optimizer) to execute queries efficiently.
 - Analyze should be run on a schedule to keep statistics up to date with the changing data. Otherwise query performance will degrade over time.

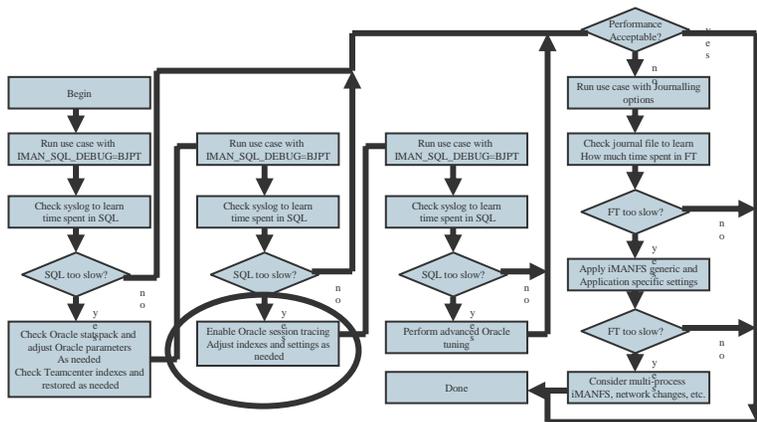
Demonstration

- Repeat I-DEAS Open test case
 - Capture the syslog and statspack reports
 - Look for remaining Oracle issues



Demonstration

- Oracle performance is still not acceptable
 - Indications of an indexing problem in the syslog and statspack report
- Enable Oracle session tracing



Demonstration

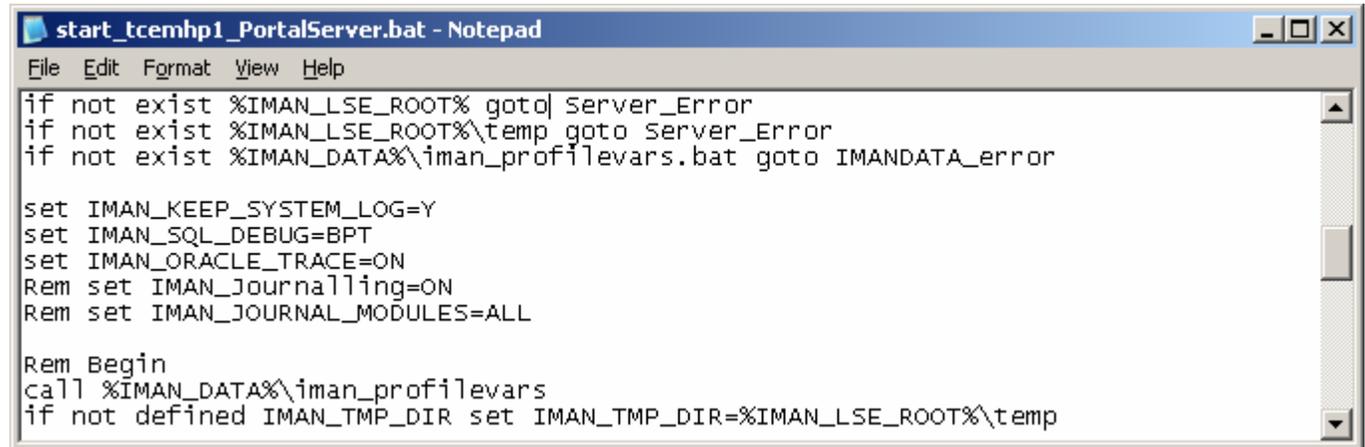
- Oracle session tracing
 - What does it tell you?
 - Explain plan for each query that Teamcenter sends to Oracle
 - Shows whether Oracle is doing a full table scan, indicating a missing index
 - Time spent in Oracle CPU for each query
 - Time spent in different wait events, per query and per session
 - To identify SQLnet or disk I/O bottlenecks
 - Session tracing is a way to see what happens on the Oracle server for a specific Teamcenter session only

Demonstration

- Ways to enable Oracle session tracing
 - Teamcenter configuration variable
 - Will show explain plans and execution times for queries, but not information about wait events
 - Direct SQL
 - Most reliable way to get a complete trace, including wait information
 - Can also use IMAN_SLOW_SQL
 - But, this shows explain plan only for queries that exceed the specified time
 - No wait information

Demonstration

- Ways to enable Oracle session tracing
 - Teamcenter configuration variable
 - Set IMAN_ORACLE_TRACE=ON in Teamcenter startup file



```
start_tcemhp1_PortalServer.bat - Notepad
File Edit Format View Help
if not exist %IMAN_LSE_ROOT% goto Server_Error
if not exist %IMAN_LSE_ROOT%\temp goto Server_Error
if not exist %IMAN_DATA%\iman_profilevars.bat goto IMAN_DATA_error

set IMAN_KEEP_SYSTEM_LOG=Y
set IMAN_SQL_DEBUG=BPT
set IMAN_ORACLE_TRACE=ON
Rem set IMAN_Journaling=ON
Rem set IMAN_JOURNAL_MODULES=ALL

Rem Begin
call %IMAN_DATA%\iman_profilevars
if not defined IMAN_TMP_DIR set IMAN_TMP_DIR=%IMAN_LSE_ROOT%\temp
```

- Output is written to Oracle udump directory, can be processed with tkprof for readability

Demonstration

- Ways to enable Oracle session tracing
 - Direct SQL
 - Find the session serial number from OEM or from the v\$session table
 - Find the process ID from the v\$process table
 - Connect to session with oradebug and enable trace
 - Trace ends automatically when session ends
 - Process output file with tkprof
 - This is the most reliable way to get a complete report. A simplified procedure is available, but will sometimes fail to generate the wait statistics

Demonstration

- Switch on Oracle session trace
 - Find the SID (Serial ID) of your imanserver session

```
C:\> Shortcut to cmd.exe - sqlplus /nolog

C:\WINDOWS\system32>sqlplus /nolog
SQL*Plus: Release 9.2.0.4.0 - Production on Thu Apr 13 08:21:23 2006
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

SQL> connect /as sysdba
Connected.
SQL> select sid, program from v$session;

   SID PROGRAM
-----
      1 ORACLE.EXE
      2 ORACLE.EXE
      3 ORACLE.EXE
      4 ORACLE.EXE
      5 ORACLE.EXE
      6 ORACLE.EXE
      7 sqlplus.exe
      8 imanserver.exe

8 rows selected.

SQL> _
```

SID = 8

Demonstration

- Find the process pid of your Oracle session
 - Use SQL as shown, after you have determined the SID

```
C:\ Shortcut to cmd.exe - sqlplus /nolog
SQL*Plus: Release 9.2.0.4.0 - Production on Thu Apr 13 08:21:23 2006
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
SQL> connect /as sysdba
Connected.
SQL> select sid, program from v$session;

  SID PROGRAM
-----
     1 ORACLE.EXE
     2 ORACLE.EXE
     3 ORACLE.EXE
     4 ORACLE.EXE
     5 ORACLE.EXE
     6 ORACLE.EXE
     7 sqlplus.exe
     8 imanserver.exe

8 rows selected.

SQL> select s.sid, p.pid from v$session s, v$process p
  2  where s.paddr = p.addr and s.sid = 8;

  SID      PID
-----
     8         9

SQL>
```

Pid = 9

*Note: This is Windows example.
For UNIX, substitute p.spid for
p.pid*

Appendix 1: Alternative Oracle session tracing procedures

- Connect to imanserver's Oracle session with oradebug, and start session trace

```
SQL> connect /as sysdba
```

```
Connected.
```

```
SQL> oradebug setorapid 9;
```

```
Windows thread id: 5520, image: ORACLE.EXE
```

- Start session trace

```
SQL> oradebug event 10046 trace name context forever, level 8;
```

Note: This is Windows example. For UNIX, substitute

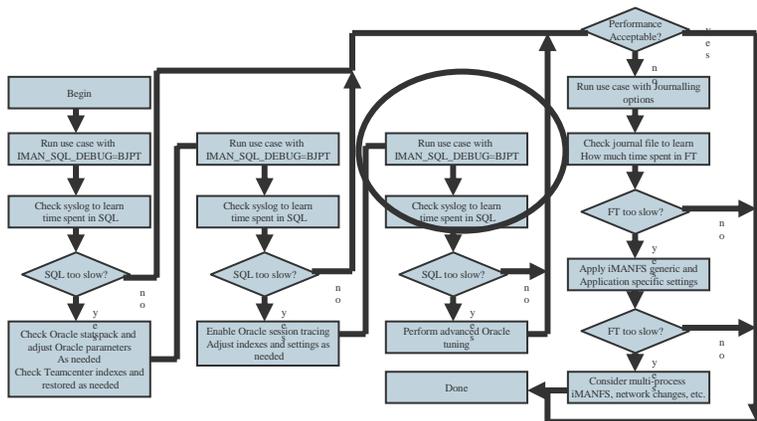
oradebug setospid

For

oradebug setorapid

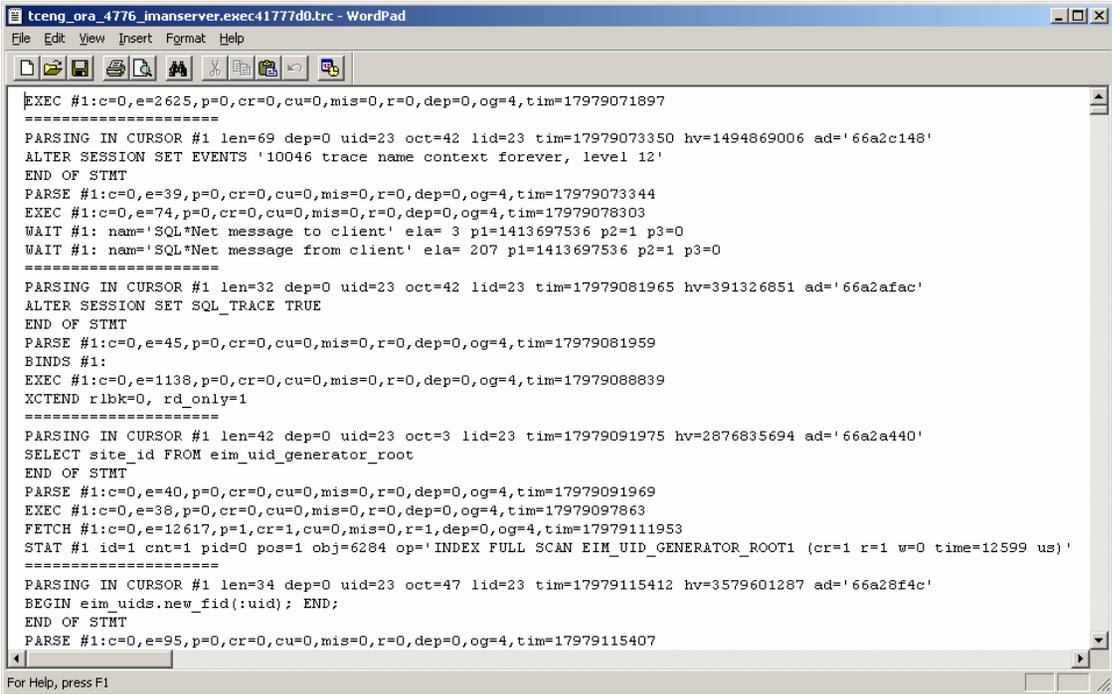
Demonstration

- Repeat I-DEAS Open test
- Analyze Oracle session trace and Teamcenter syslog, to identify missing custom index(es)
- Install custom index(es)
- Rebuild Oracle statistics



Demonstration

- Analyze Oracle session trace
 - In its raw form, session trace is difficult to read



The screenshot shows a WordPad window titled "tceng_ora_4776_imanserver.exec41777d0.trc - WordPad". The window contains the following text, which is an Oracle session trace:

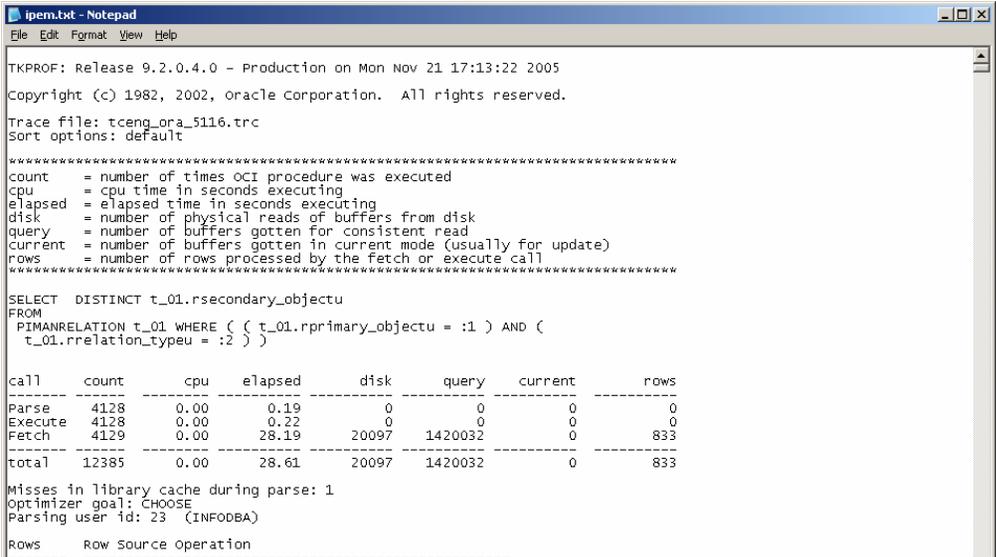
```
EXEC #1:c=0,e=2625,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=4,tim=17979071897
=====
PARSING IN CURSOR #1 len=69 dep=0 uid=23 oct=42 lid=23 tim=17979073350 hv=1494869006 ad='66a2c148'
ALTER SESSION SET EVENTS '10046 trace name context forever, level 12'
END OF STMT
PARSE #1:c=0,e=39,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=4,tim=17979073344
EXEC #1:c=0,e=74,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=4,tim=17979078303
WAIT #1: nam='SQL*Net message to client' ela= 3 p1=1413697536 p2=1 p3=0
WAIT #1: nam='SQL*Net message from client' ela= 207 p1=1413697536 p2=1 p3=0
=====
PARSING IN CURSOR #1 len=32 dep=0 uid=23 oct=42 lid=23 tim=17979081965 hv=391326851 ad='66a2afac'
ALTER SESSION SET SQL_TRACE TRUE
END OF STMT
PARSE #1:c=0,e=45,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=4,tim=17979081959
BINDS #1:
EXEC #1:c=0,e=1138,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=4,tim=17979088839
XCTEND rlbk=0, rd_only=1
=====
PARSING IN CURSOR #1 len=42 dep=0 uid=23 oct=3 lid=23 tim=17979091975 hv=2876835694 ad='66a2a440'
SELECT site_id FROM eim_uid_generator_root
END OF STMT
PARSE #1:c=0,e=40,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=4,tim=17979091969
EXEC #1:c=0,e=38,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=4,tim=17979097863
FETCH #1:c=0,e=12617,p=1,cr=1,cu=0,mis=0,r=1,dep=0,og=4,tim=17979111953
STAT #1 id=1 cnt=1 pid=0 pos=1 obj=6284 op='INDEX FULL SCAN EIM_UID_GENERATOR_ROOT1 (cr=1 r=1 w=0 time=12599 us)'
=====
PARSING IN CURSOR #1 len=34 dep=0 uid=23 oct=47 lid=23 tim=17979115412 hv=3579601287 ad='66a28f4c'
BEGIN eim_uids.new_fid(:uid); END;
END OF STMT
PARSE #1:c=0,e=95,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=4,tim=17979115407
```

At the bottom of the window, there is a status bar that reads "For Help, press F1".

Demonstration

- Analyze Oracle session trace
 - Process with tkprof to format file for readability

tkprof tceng_ora_5520.trc tkprof.txt explain=infodba/infodba waits=yes



```
ipem.txt - Notepad
File Edit Format View Help

TKPROF: Release 9.2.0.4.0 - Production on Mon Nov 21 17:13:22 2005
Copyright (c) 1982, 2002, oracle corporation. All rights reserved.

Trace file: tceng_ora_5116.trc
Sort options: default

*****
count  = number of times OCI procedure was executed
cpu    = cpu time in seconds executing
elapsed = elapsed time in seconds executing
disk   = number of physical reads of buffers from disk
query  = number of buffers gotten for consistent read
current = number of buffers gotten in current mode (usually for update)
rows   = number of rows processed by the fetch or execute call
*****

SELECT DISTINCT t_01.rsecondary_objectu
FROM
  PIMANRELATION t_01 WHERE ( ( t_01.rprimary_objectu = :1 ) AND (
    t_01.rrelation_tyeu = :2 ) )

call      count      cpu      elapsed      disk      query      current      rows
-----
Parse     4128         0.00       0.19         0          0          0          0
Execute  4128         0.00       0.22         0          0          0          0
Fetch    4129         0.00      28.19       20097     1420032      0          833
total    12385         0.00      28.61       20097     1420032      0          833

Misses in library cache during parse: 1
Optimizer goal: CHOOSE
Parsing user id: 23 (INFODBA)

Rows      Row Source Operation
-----
```

Demonstration

- Oracle session trace
 - First, jump to the bottom to see how much elapsed time is spent in processing queries

Number of queries

Elapsed time

```
nxmi_cki.txt - Notepad
File Edit Format View Help

OVERALL TOTALS FOR ALL NON-RECURSIVE STATEMENTS
call      count      cpu      elapsed    disk      query      current    rows
-----
Parse     18475      1.31     1.55       0         2214       0         0
Execute   18475      2.56     24.16     1580      18699      68625     12962
Fetch     7843       0.62     3.76       258       20237      0         8581
total    44793      4.50     29.49     1838     41150     68625     21543

Misses in library cache during parse: 144

Elapsed times include waiting on following events:
Event waited on-----Times waited-----Max. wait-----Total waited-----
SQL*Net message to client      22350      0.00      0.07
SQL*Net message from client    22350     53.88    155.94
db file sequential read        1735       0.60     24.18
SQL*Net more data from client    56         0.00      0.00
log file sync                   3830       0.17      7.28
db file scattered read          19         0.05      0.25
SQL*Net more data to client     6          0.00      0.00
```

Demonstration

- Oracle session trace
 - Is elapsed time a large percentage of SQL time shown in syslog?

```
nxmi_cki.txt - Notepad
File Edit Format View Help

OVERALL TOTALS FOR ALL NON-RECURSIVE STATEMENTS

call      count      cpu      elapsed      disk      query      current      rows
-----
Parse     18475      1.31     1.55         0         2214       0           0
Execute  18475      2.56     24.16       1580      18699     68625      12962
Fetch     7843      0.62     3.76        258      20237     0           8581
-----
total    44793      4.50     29.49       1838     41150     68625     21543

Misses in library cache during parse: 144

Elapsed times include waiting on following events:
Event waited on                      Times    Max. Wait    Total Waited
-----
SQL*Net message to client              22350         0.00          0.07
SQL*Net message from client            22350        53.88        155.94
db file sequential read                 1735         0.60          24.18
SQL*Net more data from client           56           0.00           0.00
log file sync                           3830         0.17           7.28
db file scattered read                   19           0.05           0.25
SQL*Net more data to client              6           0.00           0.00
```

```
imanserver.exe464cac0.syslog - Notepad
File Edit Format View Help

231 1 0.000 0.000 0.000 64 0 SELECT PAM_ACE.puid, ptimestamp, ppid, rownum
232 1 0.000 0.000 0.000 2 0 SELECT PACCESSORS.puid, pseq, pvalu_0, pvalc_0, pt
233 1 0.000 0.000 0.000 1 0 SELECT DISTINCT t_0.puid FROM PPOM_USER t_0 w
234 1 0.000 0.000 0.000 1 0 SELECT DISTINCT t_0.puid FROM PPOM_GROUP t_0
235 1 0.000 0.000 0.000 1 0 SELECT DISTINCT t_0.puid FROM PCLIENTVALUES t_
236 1 0.047 0.047 0.047 1 0 SELECT DISTINCT t_0.puid FROM PCLIENTDATAMAST
237 1 0.000 0.000 0.000 1 0 SELECT COUNT(*) "number" FROM POM_F_LOCK WHERE
238 1 0.016 0.016 0.016 1 0 SELECT DISTINCT t_03.puid FROM PPOM_ACCESSOR
239 1 0.000 0.000 0.000 1 0 SELECT DISTINCT t_02.puid FROM PGROUP t_02 ,
240 1 0.000 0.000 0.000 4 0 SELECT DISTINCT t_01.rAccessor_idc FROM PAM_J
241 1 0.000 0.000 0.000 0 0 OPEN CURSOR
242 1 0.000 0.000 0.000 1 0 INSERT INTO PPOM_SESSION ( puid,prod_name,pu
243 1 0.000 0.000 0.000 1 0 INSERT INTO PM_PROCESS_LIST (pm_process_id, p
244 1 0.032 0.032 0.032 37 0 INSERT INTO PCONTENTS (puid,pseq,pvalu_0,pvalc
245 1 0.000 0.000 0.000 1 0 DELETE FROM PPOM_SESSION WHERE puid = :1
246 1 0.000 0.000 0.000 1 0 DELETE FROM PPOM_OBJECT WHERE puid = :1
247 1 0.000 0.000 0.000 5 0 DELETE FROM POM_M_LOCK WHERE process = :1
248 1 0.016 0.016 0.016 0 0 DELETE FROM POM_M_LOCK WHERE process = :1
249 1 0.000 0.000 0.000 0 0 DELETE FROM POM_B_LOCK WHERE process = :1
250 1 0.000 0.000 0.000 0 0 DELETE FROM POM_BACKPOINTER WHERE from_uid = :
251 1 0.000 0.000 0.000 1 0 DELETE FROM PM_PROCESS_LIST WHERE pm_process_
252 1 0.078 0.078 0.078 0 0 CONNECT
253 1 0.000 0.000 0.000 1 0 BEGIN em_uids.new_fid(:uid); END;

Sum: 2060 49.586 27181 0

COMMIT:
err=0 rows=1 0.000 secs:
**** (commit) DB Transaction time was 0.016000 sec (cpu 0.000000)
0 seconds to POM DB x-act
000 End of session Thu Apr 13 08:46:55 2006 central daylight time
```

- In this case about 60% of the time is spent in Oracle server, processing queries

Demonstration

- Oracle session trace
 - Look for full table scans
 - Search for “table access full”

```
nxmi_cki.txt - Notepad
File Edit Format View Help

call      count      cpu      elapsed    disk      query      current    rows
-----
Parse     38         0.01     0.02       0          0          0          0
Execute   38         0.01     0.01       0          0          0          0
Fetch     38         0.00     0.04       1          38         0          0
-----
total     114        0.03     0.08       1          38         0          0

Misses in library cache during parse: 1
Optimizer goal: CHOOSE
Parsing user id: 23

Row Source Generation
-----
TABLE ACCESS BY INDEX ROWID PTC_PROJECT (cr=1 r=1 w=0 time=38908 us)
TABLE ACCESS BY INDEX ROWID PIPPROJECTTEAM (cr=1 r=1 w=0 time=38875 us)
JOIN CARTESIAN (cr=1 r=1 w=0 time=38871 us)
TABLE ACCESS BY INDEX ROWID PTC_PROJECT (cr=1 r=1 w=0 time=38861 us)
TABLE ACCESS BY INDEX ROWID PTC_PROJECT (cr=1 r=1 w=0 time=38857 us)
INDEX RANGE SCAN PIPTC_PROJECT_2 (cr=1 r=1 w=0 time=38851 us)(object id 6478)
INDEX UNIQUE SCAN PIPPROJECTTEAM (cr=0 r=0 w=0 time=0 us)(object id 6471)
BUFFER SORT (cr=0 r=0 w=0 time=0 us)
TABLE ACCESS FULL PGROUPMEMBER (cr=0 r=0 w=0 time=0 us)
TABLE ACCESS BY INDEX ROWID PPOM_MEMBER (cr=0 r=0 w=0 time=0 us)
```

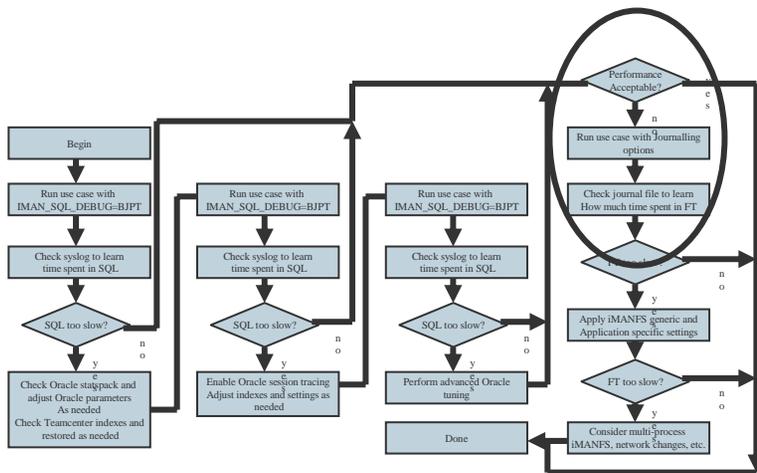
Almost always you will find full table scans, but unless the query elapsed time is a significant percentage of total elapsed time, it is probably okay

Demonstration

- Oracle session trace
 - Fill in here with pideasguids index and how to restore index

Demonstration

- After basic Oracle tuning, evaluate performance of use case
 - If performance is still unacceptable, it's time to look at other possible reasons
 - For use cases involving file transfer, the most likely candidate is iMANFS configuration

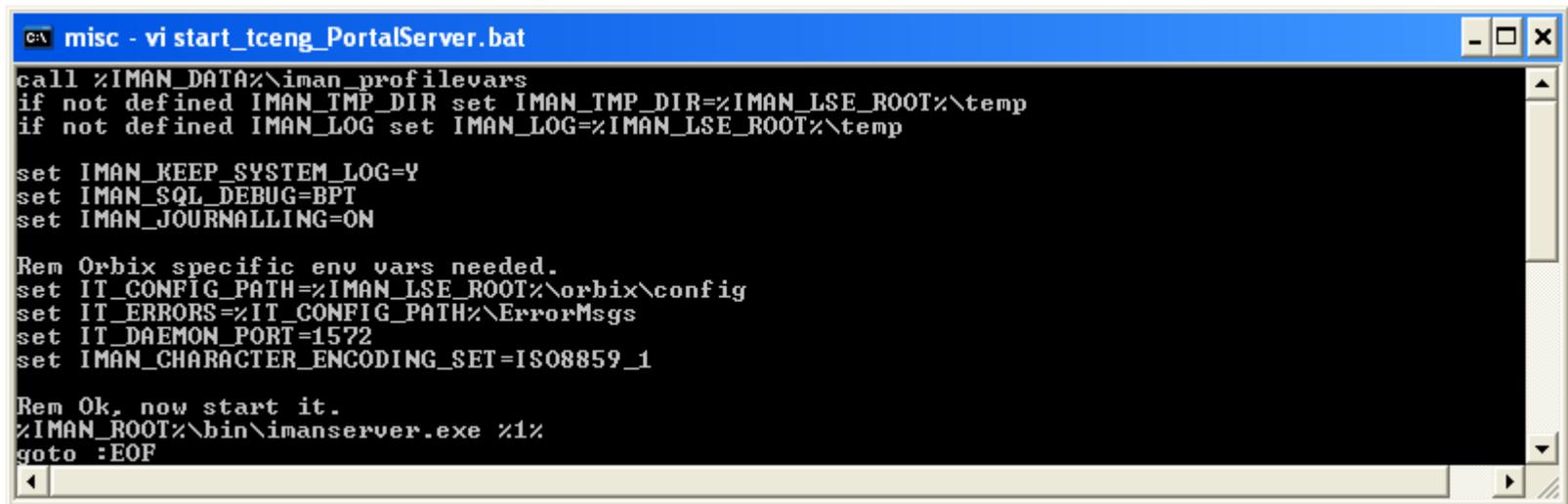


Demonstration

- **Imanserver journal file**
 - What does it tell you? It depends upon the settings used
 - Set `IMAN_Journalling=ON` to enable basic journalling
 - This will print information about high level method calls, plus summary table, for a default set of modules
 - Set `IMAN_JOURNAL_MODULES=ALL` to enable journalling for all modules
 - Use with caution: This will generate a very large journal file and will also slow all operations down very significantly
 - Set `IMAN_POM_JOURNALLING=N` to enable nested journalling
 - Use with **EXTREME** caution! The journal file will be enormous and the system will be very slow. This is only used to trace the code path for specific functions, for a limited amount of time.

Demonstration

- Imanserver journal file
 - To avoid introducing a performance degradation due to excessive journalling, use the most basic setting
 - Set IMAN_JOURNALLING=ON



```
C:\> misc - vi start_tceng_PortalServer.bat
call %IMAN_DATA%\iman_profilevars
if not defined IMAN_TMP_DIR set IMAN_TMP_DIR=%IMAN_LSE_ROOT%\temp
if not defined IMAN_LOG set IMAN_LOG=%IMAN_LSE_ROOT%\temp

set IMAN_KEEP_SYSTEM_LOG=Y
set IMAN_SQL_DEBUG=BPT
set IMAN_JOURNALLING=ON

Rem Orbix specific env vars needed.
set IT_CONFIG_PATH=%IMAN_LSE_ROOT%\orbix\config
set IT_ERRORS=%IT_CONFIG_PATH%\ErrorMsgs
set IT_DAEMON_PORT=1572
set IMAN_CHARACTER_ENCODING_SET=ISO8859_1

Rem Ok, now start it.
%IMAN_ROOT%\bin\imanserver.exe %1%
goto :EOF
```

Demonstration

- After setting journalling options, run use case and measure elapsed time
 - Remember to keep actions unrelated to the use case to an absolute minimum!
- Examine summary section of the journal file

```
imanserver.exe81fa0ca0.jnl - Notepad
File Edit Format View Help
@* (Commit) DB Transaction Time was 0.016000 sec (cpu 0.000000)
@*
@* Total session time 69.813000s, of which 67.154000s was outside journalled calls.
@* Total CPU time 6.032000s.
@*
@* Time spent in this function, or those it calls
@* % total total call average
@* elapsed elapsed cpu counts time
@*-----
@* 13 0.531 0.47 2 0.266 POM_load_class_extnt
@* 12 0.517 0.313 50 0.0103 BOM_line_ask_child_lines
@* 8.7 0.36 0.078 226 0.00159 FS
@* 5.7 0.235 0.123 186 0.00126 AOM_refresh
@* 4.5 0.188 0.096 4 0.047 FL_ask_releases
@* 4.5 0.187 0.157 2 0.0935 POM_init_module
@* 3 0.126 0.08 324 0.000389 WSOM_ask_object
@* 1.9 0.077 0.077 49 0.00157 PS_ask_occurrence
@* 1.5 0.063 0.062 71 0.000887 POM_old_execute_enquiry
@* 0.77 0.032 0.031 568 5.63e-005 POM_class_of_instance
@* 0.77 0.032 0.016 38 0.000842 RES_is_checked_out
@* 0.75 0.031 0.016 1 0.031 AE_init_module
@* 0.75 0.031 0.032 1 0.031 POM_start
@* 0.75 0.031 0.031 79 0.000392 ATTRMAP_find_mapping
@* 0.75 0.031 0.015 1 0.031 SA_init_module
@* 0.39 0.016 0.015 549 2.91e-005 POM_class_id_of_class
@* 0.39 0.016 0.016 196 8.16e-005 BOM_line_look_up_attribute
@* 0.39 0.016 0.016 1 0.016 ListofValuesIntegerP::initialize
@* 0.39 0.016 0.016 1 0.016 LOV_init_module
@* 0.39 0.016 0.015 1 0.016 POM_find_accessor
@* (+ another 168 routines)
@* 1.6e+003 67.2 4.29 4322 0.0155 Time outside journalling (total gap t
@*
@* Journal file closed on Thu Apr 13 12:57:03 2006
```

Time spent in FS module is a rough indicator of how much time is spent transferring files

Demonstration

- How do you know whether file transfer is too slow?
 - How much time to transfer the files via ftp?
 - Divide file size by bandwidth
 - For example, on a slow WAN the bandwidth might be around 50 Kbytes/second. To transfer an 8MB assembly would not take less than 163 seconds
 - $(8 * 1024) / 50 = 163$ seconds

```
imanserver.exe24be6860.jnl - Notepad
File Edit Format View Help
@* Total CPU time 10.422000s.
@*
@* Time spent in this function, or those it calls
@* % total total call average
@* elapsed elapsed cpu counts time
@* -----
@* 83 745 1.15 4225 0.176 FS
@* 4.2 37.4 0.219 322 0.116 WSOM_ask_object_type
@* 2.8 25 0.33 50 0.499 BOM_line_ask_child_lines
@* 2.8 24.9 0.236 186 0.134 AOM_refresh
@* 1.4 12.2 0.031 54 0.226 RES_ask_reservation_object
@* 0.96 8.62 0.063 49 0.176 PS_ask_occurrence_note_text
@* 0.66 5.89 0.031 1 5.89 POM_start
```

Need to check iMANFS,
since FS time is so much
larger than FTP

Demonstration

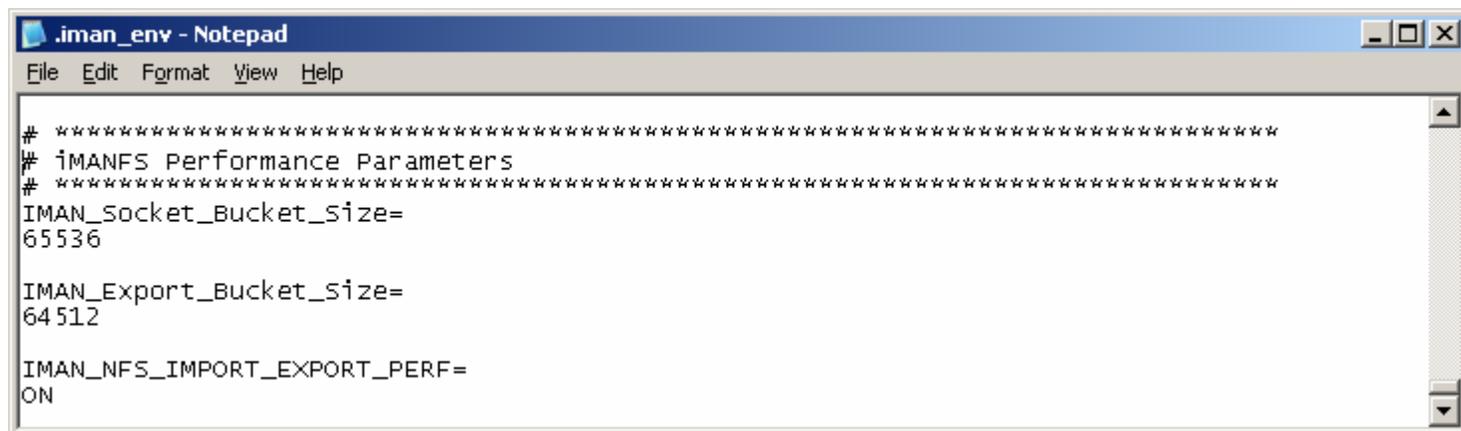
- iMANFS parameters
 - Some parameters depend on what mode of file transfer is used
 - To determine FT mode, check the syslog
 - IMANVolume::load tcemhp1: accesses in mode TCP (iMANFS mode 3)
 - When in TCP mode
 - Set IMAN_Export_Bucket_Size=64512
 - Default if not set is 10k
 - Controls how size of each block transferred by iMANFS
 - Set IMAN_Socket_Bucket_Size=65536
 - Default if not set is system default
 - Controls the socket window size

Demonstration

- iMANFS parameters
 - When in CIFS mode
 - Set IMAN_NFS_IMPORT_EXPORT_PERF=ON
 - Controls how IMAN_Export_Bucket_Size setting is interpreted
 - Only effective for certain clients!
 - Set IMAN_Export_Bucket_Size=64512
 - Default if not set is 128k
 - Controls how size of each block transferred by iMANFS

Demonstration

- iMANFS parameters
 - Recommended settings to cover all possibilities



```
.iman_env - Notepad
File Edit Format View Help
# *****
# iMANFS Performance Parameters
# *****
IMAN_Socket_Bucket_Size=
65536

IMAN_Export_Bucket_Size=
64512

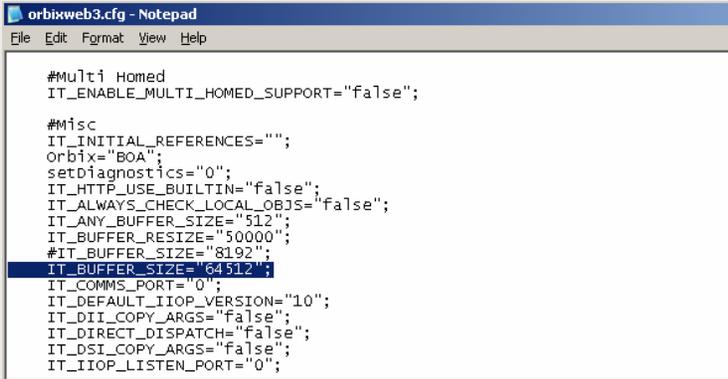
IMAN_NFS_IMPORT_EXPORT_PERF=
ON
```

- Some can be set in environment, but for ease of maintenance you should set them in .iman_env, on both server and client

- iMANFS parameters
 - Some clients, such as NX Manager/I-DEAS and Pro/E Manager, do not recognize IMAN_NFS_IMPORT_EXPORT_PERF
 - They call a different file transfer API in imanserver
 - For such clients, you must set
 - IMANFile_transfer_buf_size=64512
 - In
 - %IPR%\com\ugsolutions\iman\kernel\kernel_user.properties
 - The default for this variable is 2048 bytes - too small

Demonstration

- iMANFS parameters
 - IT_BUFFER_SIZE: An Orbix configuration parameter that controls the size of the buffer used for marshalling operation parameters
 - Default value is 8192 bytes, which is unnecessarily small for Teamcenter
 - Change to 64512 in the %IPR%\Orbix\config\orbixweb3.cfg file



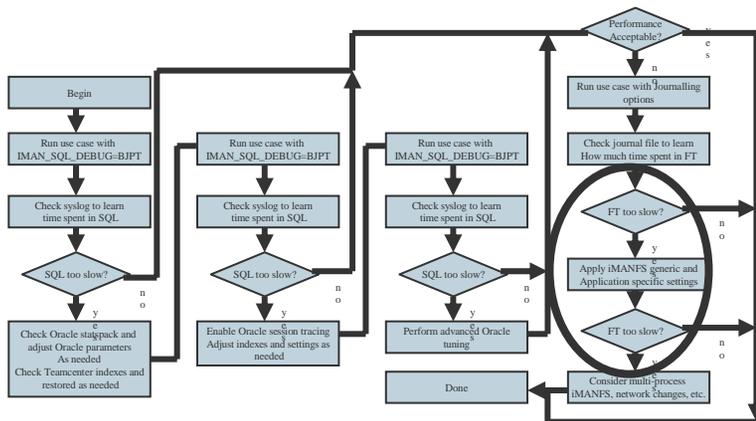
```
orbixweb3.cfg - Notepad
File Edit Format View Help

#Multi Homed
IT_ENABLE_MULTI_HOMED_SUPPORT="false";

#Misc
IT_INITIAL_REFERENCES="";
Orbix="BOA";
setDiagnostics="0";
IT_HTTP_USE_BUILTIN="false";
IT_ALWAYS_CHECK_LOCAL_OBJS="false";
IT_ANY_BUFFER_SIZE="512";
IT_BUFFER_RESIZE="50000";
#IT_BUFFER_SIZE="8192";
IT_BUFFER_SIZE="64512";
IT_COMMS_PORT="0";
IT_DEFAULT_IIOB_VERSION="10";
IT_DII_COPY_ARGS="false";
IT_DIRECT_DISPATCH="false";
IT_DSI_COPY_ARGS="false";
IT_IIOB_LISTEN_PORT="0";
```

Demonstration

- Apply iMANFS settings and rerun test case
- ☑ ▪ IMAN_Export_Socket_Size=64512
- ☑ ▪ IMAN_Socket_Bucket_Size=65536
- ☑ ▪ IMAN_NFS_IMPORT_EXPORT_PERF=ON
- ☑ ▪ IMANFile_transfer_buf_size=64512
- ☑ ▪ IT_BUFFER_SIZE=64512



Demonstration

- Verify larger iMANFS block size by number of readblks in the log file
 - Before (in this example total file size is 817k):

```
imandba@moshp1$ grep readblk imanfs27086.log | wc -l
```

419
 - $817/419 = \sim 2 \Rightarrow$ files are read from volume in 2k blocks

After:

```
imandba@moshp1$ grep readblk imanfs27178.log | wc -l
```

30

$817/30 = \sim 27 \Rightarrow$ files are now read from volume in 27k blocks

Additional Techniques

- Now we are done with basic Oracle and iMANFS performance tuning
- If performance is still not acceptable, then additional steps may be required
 - Oracle configuration
 - Disk defragmentation
 - Network changes
 - Teamcenter configuration

Additional Techniques

- Oracle configuration
 - Tablespace and control files
 - Distribute across disks to eliminate I/O bottlenecks
 - User tables and indexes
 - By default these are created in IDATA tablespace
 - Certain tables such as POM_BACKPOINTER are accessed so frequently that there is I/O contention on the IDATA tablespace file
 - Move frequently-accessed tables and indexes into new tablespaces, with files located on separate disks, to minimize I/O bottleneck
 - These procedures are documented in the performance checklist, but typically should be done by a DBA

Additional Techniques

- Teamcenter configuration
 - Multi-process iMANFS
 - If single-user performance is not acceptable then this isn't going to help; only if significant degradation occurs under concurrent loading
 - Three tier configuration
 - Several CAD managers support three tier deployment; performance across a WAN may improve
 - Application-specific capabilities
 - File caching
 - APB (for NX/Manager)
 - Operating system configuration
 - Especially Windows – Oracle memory issues

Resources for field performance tuning

- Documentation
 - Performance checklist
 - Streamlined version of Deployment Guide, with some information about undocumented settings
 - Instructions for using statspack, moving Oracle control files and tablespaces, etc.
 - Deployment Guide
 - Comprehensive manual updated for each major release
 - Download from http://support.ugs.com/docs/tc_eng.shtml
 - Oracle statspack parser
 - <http://www.oraperf.com/>
 - Upload your statspack file and it will generate a summary of performance issues and recommendations

Summary

- Informal methodology for identifying/resolving the most commonly seen performance problems at production sites
- If further help from GTAC or Product Development is needed, please provide
 - Use case details
 - Test data set (if possible)
 - Imanserver syslog and journal files
 - Oracle statspack report
 - Oracle session trace, run through tkprof