TEMENOS™

# T24 PERFORMANCE

# - A TUNING GUIDE



Information in this document is subject to change without notice.

No part of this document may be reproduced or transmitted in any form or by any means,
Electronic or mechanical, for any purpose, without the express written permission of TEMENOS Holdings NV.

Copyright 2002-2003 TEMENOS Holdings NV. All rights reserved.

**CONTENTS**

Document History:

| Author | Version | Date |
|---|---|---|
| Tony Coleman | 1.0 | 10th Jun 2008 |
| P&S | 2.0 | 15th Jun 2008 |
| P&S | 3.0 | 2nd Dec 2008 |
| Mutu | 4.0 | 2nd May 2012 |

| Comments: |
|---|
| Initial Version |
| Added Tuning changes done in Pzero for T24 Release R08. |
| Minor Corrections and formatting changes. Added Doc History. |

# INTRODUCTION

## Purpose

This document aims to provide an approach for monitoring and tuning performance for T24, and indeed any IT system. It is not intended to be an exhaustive guide to all the possible performance issues that may arise, but is aimed at tutoring a methodical approach to identify the factors that are limiting performance, and the steps that can be taken to mitigate the effects of these limitations, or to remove them entirely. Although this document intends to give high level monitoring and performance details for the various components of T24 it is not intended to be a comprehensive guide for all components. The relevant product user guides should be consulted.

The goals of performance and rich business functionality are opposed. Performance issues that arise in T24 may be a result of the parameters that are shipped with the application – these offer the most detailed business functionality – but at a cost. At time of writing, the T24 Model Bank is actively being modified to set the default settings to be performance oriented. It should be noted that many, if not all, parameter settings for other software (including operating systems) are NOT shipped with performance oriented defaults. It all needs tuning. This document can ever truly be "finished". T24 and the IT components around it are constantly evolving.....

## Terminology

| | |
|---|---|
| T24 | Refers to the T24 system as a whole |
| T24 Browser | The T24 browser based user interface. Refers to both the servlet and J2EE components |
| Application Server | The software (or machine) that hosts the J2EE container. This is an industry term and does not refer to the T24 business logic |
| T24 Server | The server where the T24 business logic runs |
| Jbase | The run time environment for the T24 Server |
| J4 | The default database for the T24 Server |
| UPM | Universal Performance Metrics |
| TOCF (EE) | Temenos Open connectivity Framework – Enterprise Edition |
| JMS | |
| TAFC | |
| COB | Close of Business |
| TCS | Temenos Connector Server |
| TCC | Temenos Connector Client |
| SSL | Secured Socket Layer |
| AWR | Automatic Workload Repository |
| ADDM | Automatic Database Diagnostic Monitor |
| DBA | Database administrator |

**Formatted:** Font: (Default) Arial
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt
**Formatted:** Font: (Default) Arial, 10 pt

## Types of Performance Limitations

All IT systems will eventually become bound by some limitation. In terms of this document the discussions will be limited to the following types of performance binding:

- Lock bound
- IO bound
- Network Bound
- CPU bound
- Memory bound

### Lock Bound

Systems can become lock bound when more than one process tries to update the same row in a table at the same time. When this happens, processes must wait for the lock to be released before continuing. While it is necessary to lock these items, lock contention can be avoided by careful presentation of data to the system ( i.e. sort a clearing file so that the entries are not in account number order!). An example of a situation that may cause a system to be lock bound is poor application design (where the system is required to update a row on a table with a generic key, i.e. currency or date) or where the business data forces the application to update a single account. In the latter case, there is functionality within T24 to solve this issue.

### IO Bound

Systems can become IO bound when the amount or variety of IO exceeds the hardware device's performance. Typically, this is most prevalent on log files on databases. Most systems should be mirrored and striped, using either RAID 01 or RIAD 5. IO biding may also occur where there is unnecessary IO operations being performed. This can be resolved either by parameterisation or by code changes. Faster disks, will (of course) help.

### Network Bound

Systems become Network bound when the network traffic comes close to or exceeds the available bandwidth. Typically, this is caused by large messages (i.e. XML can be very verbose) or by a very high number of messages. The latter is usually only evidenced between application servers and database servers, while the former is more prevalent from web server to application server. If the size or number of messages cannot be reduced, then more network bandwidth will solve the problem. Typically, this will require an application change. If the Client's architecture is 3 tier then network connection between each tier should be checked for bandwidth limitation. One of main things to check for network connection is to check the network kernel level settings (ex: tcp buffer size, time wait etc)

### CPU Bound

Systems become CPU bound when the available processing power is insufficient to perform the required tasks. Most systems exhibit three types of CPU usage: user, system and wait. User time is the time taken up by the application running in the server. Wait time indicates that the processes is waiting on IO (may be disk or network) while system time indicates everything else – locks, process switching, etc. Systems typically become CPU bound when processes "spin" and are out of control, or spend a lot of time in "system" time. Killing the offending processes should relieve the situation, but it is obviously better to avoid the situation entirely. Certain circumstances can trigger jbase processes to spin, and there are environment variables that can be used to counter this. Monitoring the processes to determine where they are spending the processing time can indicate bad code that should be amended to reduce CPU time.

Formatted: Font: (Default) Arial

Formatted: Font: (Default) Arial, 10 pt

Formatted: Font: (Default) Arial, 10 pt

Formatted: Font: (Default) Arial, 10 pt

Formatted: Font: (Default) Arial, 10 pt

Formatted: Font: (Default) Arial, 10 pt

Formatted: Font: (Default) Arial, 10 pt

An example that might lead to excessive CPU activity and ultimately binding the system limitation due to CPU power is Enquiries that are poorly designed (i.e. running from STMT.ENTRY instead of relevant index tables) which leads to numerous and expensive SELECT statements being issued.

Formatted: Heading 3

## Dummies Guide to Networks

It is important to note that while the most often used metric when discussing networks is bandwidth; this is not the critical aspect of network performance. The key aspect is the response time, or latency, of the network. Network response seriously degrades once bandwidth usage exceeds 50% of the available bandwidth.
Network bandwidth is the Maximum bit rate per second that a logical or physical computer network can process. Network bandwidth is quoted in bits, i.e. 1 gigabit. However, most monitoring tools report usage in terms of bytes. Hence, a 1 gigabit network offers 128 Megabytes of bandwidth, of which 64 Megabytes can be used in a performing manner. This sounds a lot, but if a typical message is 4 Kb, then this network can only support 30 thousand per second.

$$128 * 1024 / 4 = 32,768$$

This sounds impressive, but when processing thousands of transactions per second it is easy to exceed this number in terms of messages from application server to database server. Indeed, if you assume that a typical transaction will perform 10 IO operations (which is comparatively light) then a dedicated 1 gigabit network from application server to database server will only allow some 3,000 transactions per second. And again, whilst this sounds impressive for online transactions, for close of business processing this is not sufficient for high volume sites.
Of particular note are the default kernel settings that most operating systems ship with. The key settings of the number of connections, packet size, etc. are set terrifyingly low and often include settings designed to prevent denial of service attacks against web servers. These settings are critical and need to be modified in order to utilise the available bandwidth. Just because a 1 gigabit card is installed, doesn't mean that you can use it fully!
To use an analogy, a network could be considered as a transport system. The messages are people, and the packet size is the size of the bus used to move the people. The number of connections can be seen as the number of buses.
The bigger the packet size, the more information can be moved at once, but often the "bus" must fill up before it can move. Setting the packet size too small means that your message might have to wait for a "bus", but set it too large and your message might have to wait for the bus to fill up. This analogy serves us well, as it is important to remember that the bus has to travel both ways to complete the trip – network usage is the sum of the inbound and outbound traffic, and these metrics are often reported separately. Remember to add them up.

The recommendations for network are simple:

1. **Ensure the TCP settings have been tuned on all servers**

   Eor Example the Benchmark were done with the following values to start with.
   sbmax = 1280K
   net_malloc_police = 32k
   rfc1323 = 1
   tcp_mssdflt = 1460

   tcp_nodelayack = 1

   tcp_recvspace = 256k

   tcp_sendspace = 256k

   The full lists of network kernel parameters are not in the scope of this document.

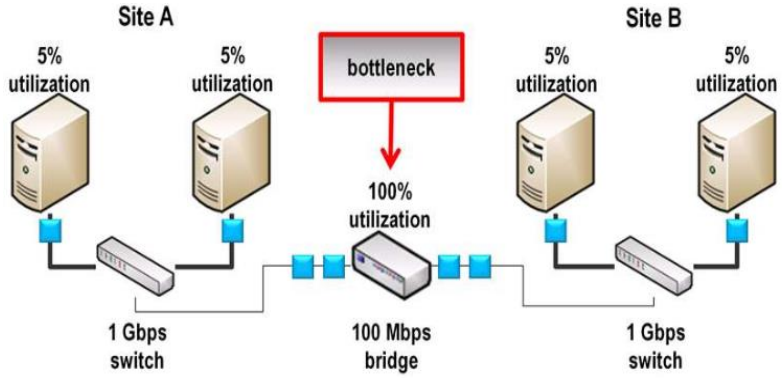
2. **Run a network performance test independent of T24**

**Run UPM-Net between the Web server to App server and appserver to DB server.**

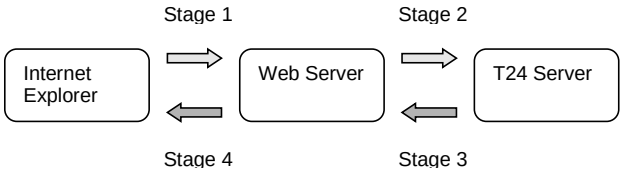
3. **Add more network cards if there is a bottleneck**

**Network throughput is only as fast as the slow link.**

As in the below diagram the connectivity between Site A and Site B is limited by the slower 100mbps device and the combined traffic on both sides. Also note that network statistics from a single computer might not reveal the network congestion.




When it comes to T24 architecture in general, any message-response communication between a T24 Browser user and T24 goes through four stages or steps. Two of these steps are to submit the user's message to T24 and the final two are to receive T24's response to the user.
Any request for a version (user screen), transaction commit, or enquiry, is a message-response that must complete these four steps.




| Stage | From | To | Communication |
|---|---|---|---|
| 1 | Internet Explorer | Web server | HTTP request |
| 2 | Web server | T24 | XML request compressed |
| 3 | T24 | Web server | XML response compressed |
| 4 | Web server | Internet Explorer | HTTP response |

Therefore to execute a complete business transaction such as a Funds Transfer (FT), eight stages are in fact required for the entire roundtrip transaction.
Four of these would be to request & display the FT version, and a further four to commit the completed transaction and return T24's confirmation to the user.
So to conclude, if there is a problem in the network, it is necessary to analyse the network efficiency and the traffic between the different layers mentioned above ( IE to Web server & Web server to T24 App server).
If the databases separated by a network, same procedure is to be applied to find the network issues.

**Output queue length** is one of the main parameter to be checked for network to be healthy. A sustained number of packets in the queue might be due to many reasons – but if this condition is reached then, it requires further investigation on the network.

## Dummies Guide to IO

Today's disk systems are complex. Very complex, and nearly always consist of multiple physical disks that are striped and mirrored with a large caching mechanism. The operating system, and hence the database, do not see the physical disks but instead see "logical disks" which represent the "front end" of the disk system. This is the disk cache, which is essentially a lump of memory. The operating system interacts with the front end, which attempts to fulfil the IO requests and where necessary flushes the requests to the "back end" which represents the physical disks. Hence, it is entirely possible to perform IO operations without a physical read taking place.
So to use another analogy, imagine that the cache is a bath, with a tap filling up a bath with IO requests. The disks are the drain (without the plug in!) draining the water from the bath. Our IO requests are the water, and as long as the bath does not fill up, the IO request is serviced efficiently. But as the bath fills up, the requests start to take longer to fulfil. Unfortunately, this analogy fails us when the bath starts to over flow, as the IO request simply wait rather than spill out.
The key goal is to achieve a non-blocking storage architecture. The recommendations for IO are simple:
**1. Stripe all files (including log files) across all available disks using a one megabyte stripe width***
**2. Mirror data for high availability**
**3. Place frequently accessed data on the outside half of the disk drives**

* The caveat here is that the storage system has sufficient non-volatile RAM to cache write the log file writes. This is normally the case, but if not the recommendation is to use a stripe size of 64k.

If database and application are not in the same server then the IO on the db server has to be monitored mainly for IO related issues.

It is a best practice to keep the data and redo logs separately in case of Oracle database on different file systems or different disks so the I/O load does not affect the system online performance.

IOPS – I/O Operations per Second is an industry standard for measuring the hardware throughput of a disk subsystem. The response time greater than 15ms is considered slow.
Disk response times are the best indicators of disk performance and the average Disk Queue length is helpful only as supplemental information.

The I/O load to the physical disks should be distributed across all the available hard disks – if only few of the hard disks are active most of the time, then disk stripping at the file system level should be analysed.

## Dummies Guide to CPU

Many chips are now multi-core based, and this can lead to confusion when trying to rate a given chip for it processing "power". The lead metric for a CPU is the clock speed – the number of instructions that it can process per second. Multi core chips can effectively be viewed as multiple processors and may or may not give close to mathematically perfect performance, i.e. a dual core 2GHz processor may or may not perform as well as two single core processors. Different vendors will quote different "efficiencies", ranging from 60% to 100%. In practice, it has been seen to be approximately 80%, i.e. our dual core 2GHz chip gives us 80% of the performance offered by two single cores. Unfortunately, this efficiency number can vary massively, and really needs to be discovered empirically for a given chip set.

The following is an illustration will give a better understanding on the terminologies related to CPU ie. Socket, Chip, Core, threads.

**Usually the processer will have a socket to hold the processor chip** so

1 socket = 1 chip

**Each chip will contain more cores**

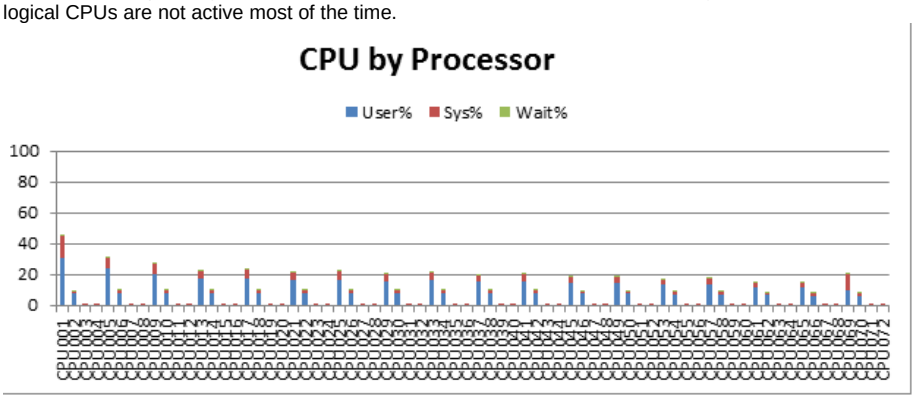
1 chip = x cores (n * cores) - For eg. Power7 Chip = 8* Power7 4Ghz Cores.

**Each core can have many SMT threads enables.**

1 core = 'nn' SMT threads. For Eg 1 Power7 Core = 4 HW SMT threads – where in AIX, when SMT is anabled each SMT is seen as a logical CPU.

1 Socket = 1 chip = 8 Cores = 32 SMT threads = 32 AIX logical CPU.

To profile the CPUs for the load during the online or COB, it is always better to profile using the CPU cores and upon monitoring the performance and CPU utilisation the no of agents can be increased or decreased. The profiling should not be based on the threads – as we have many instances where the logical CPUs are not active most of the time.




As can be seen from the logical thread utilisation, the load is not shared across all the available Logical CPUs.

An important point to look for is the number of floating point processors – effectively the "bit that does the complex maths part" – T24 performs many complex mathematical functions, and the efficiency rating of multi-core chips can reduce dramatically if there is only a single floating point processor. At time of writing, this is often the case.
You will be relieved that there is no analogy used here.
In simple terms, it is currently rare to see T24 CPU bound unless the box is terribly undersized. The solution is simple – add more CPUs, either vertically (to the same box) or horizontally (add more boxes).

While the system is in online phase or during the COB if the CPU utilisation is consistently above 85% then the server is prone to CPU issues.

# THE APPROACH

Typically, the first sign of a performance bottleneck is that adding more worker sessions does not result in a greater throughput on the target workload, i.e. T24 is seen not to scale. T24 is inherently scalable, and where scalability is not seen, then one of the five limiting factors is at play. As this is witnessed, the five factors need to be monitored to determine the current limiting factor.

In simple terms, it is a case of determining which of the five limiting factors is acting on any system under test at any given point in time. Removing that limiting factor will inevitably reveal another, the cause of which must be determined and overcome. The process repeats until the desired performance level is reached.

The overall performance of a system under test will be limited by the worst of the factors outlined above. Imagine the whole T24 system as a series of funnels, with water pouring from the top to the bottom. The flow of water will never exceed the narrowest point. When we increase the flow of a given funnel, we notice that it is another funnel that is now limiting the flow.
T24 scales up inherently both vertically and horizontally which is proved time and again on the benchmarks – (refer the BM results on the knowledgebase). To pin point the scalability issues and to check the stock of the system resources use UPM tool sets. UPM package has complete set of package for all the system components ie CPU, IO, DB, Memory and Network. Use the appropriate tool to check the system performance and scalability for the load.

# MONITORING

## Browser

### Logging

The **browserparameters.xml** file controls the logging of messages at the web server for T24 browser. The relevant section of the document is shown below.

```
<parameter>
        <parameterName>Log Events</parameterName>
        <parameterValue>YES</parameterValue>
        <!-- Options: YES / NO -->
</parameter>
<parameter>
        <parameterName>Log Level</parameterName>
        <parameterValue>DEBUG</parameterValue>
        <!-- Options: NONE / INFO / DEBUG -->
</parameter>
```

Log Events is a top level setting to turn off logging. The log level is used to control how much information is sent to the logging process:

- NONE – no information is logged
- INFO – critical information is logged, but not the content of messages
- DEBUG – full and most detailed logging

All logging from the Browser currently goes to the logfile in log4j format, plenty of 3rd party log4j analysers are available to display the log files in different methods to analyse the data.

To monitor the performance or system health for T24 browser is the log file and enabling log files might affect T24 browser performance. So only when to monitor the system it's recommended to set DEBUG option for browser log.

**Note:-** Please refer each web server monitoring section to get more information about monitoring Web servers

### TCClient Logging

The tcclientlog.properties file controls the logging details of TCClient at web server level.

log4j.rootLogger=FALSE
log4j.logger.common=INFO, file
log4j.logger.channels=INFO, file
log4j.logger.ofsml=INFO, file
log4j.logger.tcc=INFO, file

Above mentioned components are available to monitor and by default INFO level details are captured for all TCClient components. But there are plenty of levels are available to monitor based on requirements

Log Level = OFF, FATAL, ERROR, WARN, INFO, DEBUG, ALL

All logging from the TCClient currently goes to the logfile in log4j format, plenty of 3rd party log4j analysers are available to display the log files in different methods to analyse the data.

To monitor the performance or system health or to analyse the issues the log files can be used, enabling log files might affect TCClient performance and in turn T24 Browser performance. So only when necessary it's recommended to set DEBUG or ALL option for TCClient log.

Jbase_agent

Logging at the jbase_agent has the following options:

0 - Log Info

1 – Log_Warning

2 – Log Error

3 – Log Fatal

The logged data are not stored in any file, the logged data can be seen on the console where jbase_agent is running if required the logged data can be redirected to a file.

## Response time information

Each and every browser request stores response time information in the header of the xml response, and breaks down the total response time into the following sections:

- total elapsed time
- time in servlet (includes time in http->xml parser)
- Time in transport (i.e. in MQ, etc.)
- time in TSS (i.e. the time spent at the T24 server processing the message)
- time in xml parser

The display on the browser status bar is delimited by |, and is displayed as:

**Total | servlet | transport | TSS | xml parser**

In the xml it is a hyphen. Below is a fragment of a response:

```
<userDetails>
    <time>191-30-40-121-10</time>
    <company>BNK MODEL BANK 14007</company>
    <companyId>US0010001</companyId>
    <multiCo>Y</multiCo>
    <user>TONY1</user>
    <skin>default</skin>
    <helpdir/>
    <today>07/03/2003</today>
    <release>G14.2.00</release>
    <compScreen>MAIN</compScreen>
</userDetails>
...
```

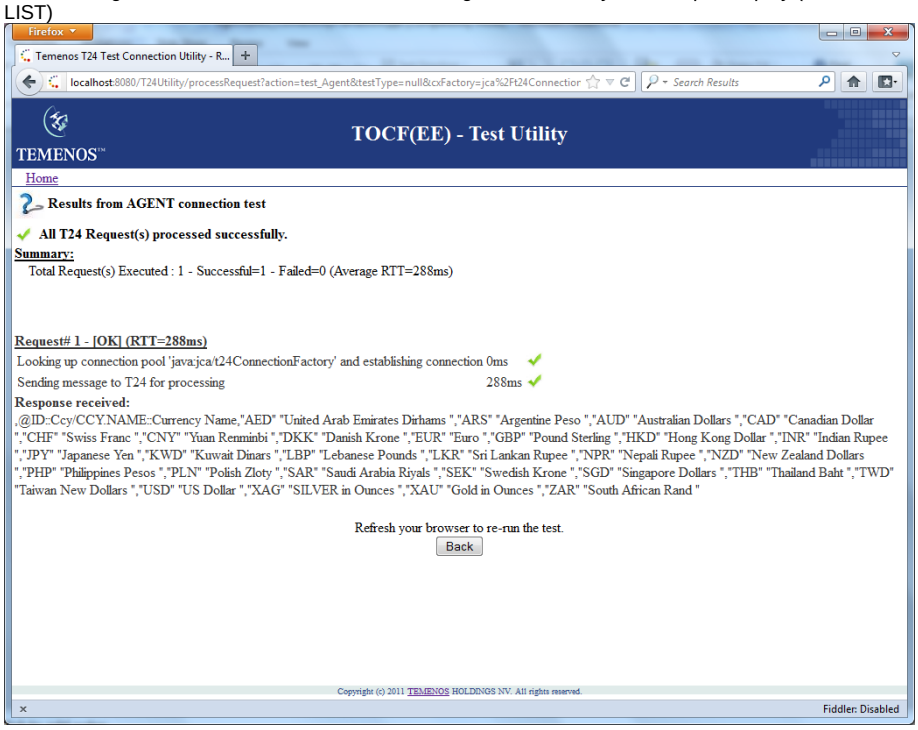
This means 191 ms in total, 30 at the servlet (of which 10 was the DOM creation of the xml message from the http request), 40 in transport and 121 in OFS (i.e. T24). The last number is included for information and breaks down the time spent in the servlet slightly.

Note this functionality works only if the TCS is configured with T24 Browser. In later releases if TOCF(EE) is deployed then the timing can be checked on the jBoss logs or to some extent with some external utilities like fiddler.

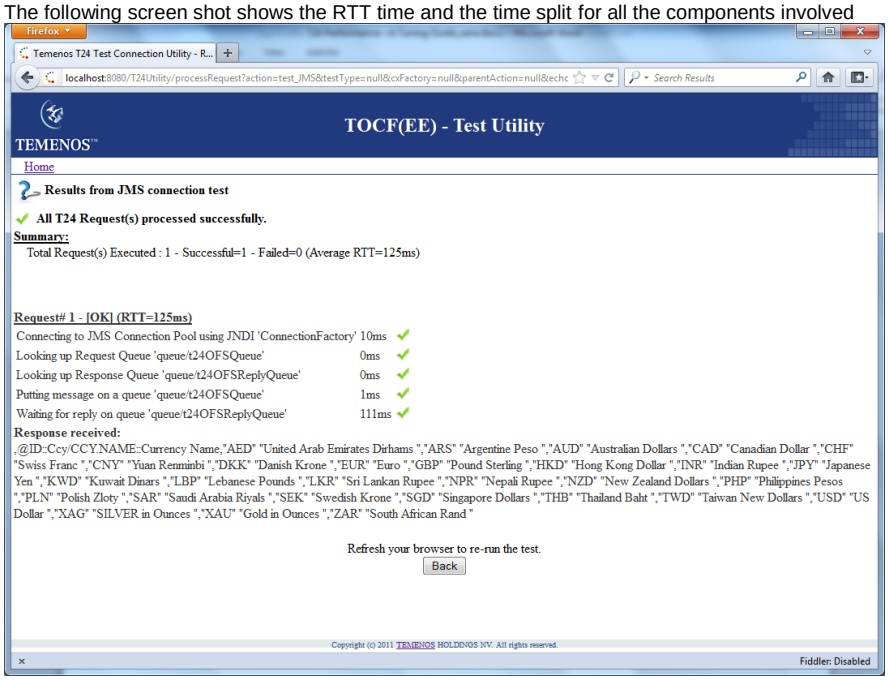
## T24Utility

This is test utility which is to be deployed on the WAS to test the response time and average Round trip time for the requests at JMS, T24ra & T24 Application. The requests can be in standard OFS or OFSML format.

The following screen shot shows the RTT time on Agent connectivity for a simple enquiry (CURRENY-LIST)



The following screen shot shows the RTT time and the time split for all the components involved



Connector - TCServer

## Log files

The **tcslog.properties** file, found under the $TCSHOME/**conf/tcserver** directory of the TCS install controls the logging process of the connector, and can be configured with various levels of reporting, to a file or to console or many other options are available.

In tcslog.properties many options are available to capture lot of useful information for monitoring TCServer performance and also T24 performance. TCS log files can also help to find any issues with connectivity with T24, error messages from T24, T24 processing time, environment setting etc

Other than log file TCServer also has a powerful monitoring tool to monitor TCServer which is called TCMonitor. A sample screen shot is shown below

## TC Monitor

The TC Monitor is a monitoring tool allowing remote connection to any TCServer. It allows the user the stop the server, check the active channels, reset the statistics data etc. Multiple monitors can be run simultaneously.

To run the TCMonitor, copy the tcmonitor.jar (which is in the lib directory of the installation path) to any desired location and start it with the script in the install directory or type “java -jar tcmonitor.jar”.

By default, the TCMonitor will try to connect to a server on « localhost », on port 9500. Connection to another server can be reached using the “File”, “Connection to …” menu options. The server is specified by ip_address_or_machine_name>:<tcp_port>. The tcp.port element is optional.

The port is listening on can be changed using the <MONITOR_PORT> The port the TCServer i er tag in the tcserver.xml file. If the <MONITOR_PORT> is not specified, the TCServ will simply not listen on any port.

TCMonitor gives the following useful information

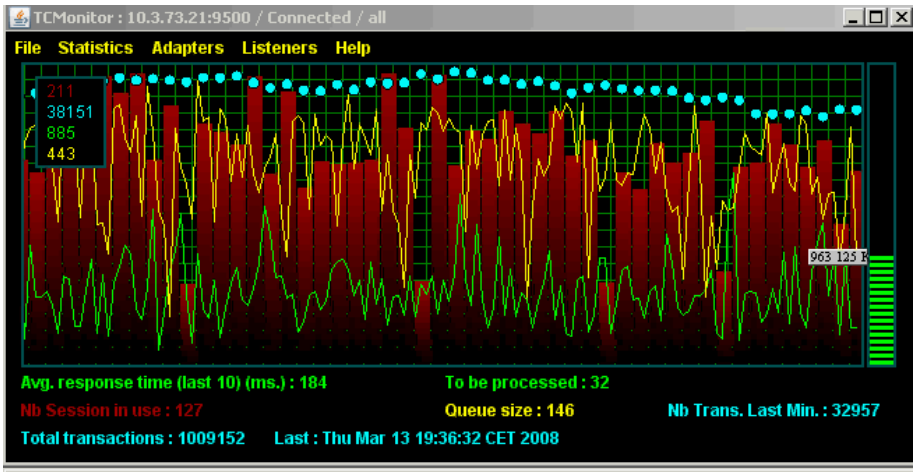
- Throughput per min
- Number of tSS sessions active and how many available
- Average time taken to process the messages
- Memory taken by TCServer
- Queue size and currently processed size which are internal parameters to TCServer
- Total transactions processed since last time monitor started

Formatted: Font: (Default) Arial, 10 pt

Formatted: Font: (Default) Arial, 10 pt

Formatted: Font: (Default) Arial, 10 pt, German (Germany)

Formatted: Font: (Default) Arial, 10 pt




## Telnet Daemon

The TCServer can be managed / monitored via a telnet session using the correct port number the port the TCServer is listening on can be changed using the <TELNETD_PORT> tag in the TCServer.xml file. If the <TELNETD_PORT> is left unspecified, the TCServer will not start the telnet deamon.

**eg : telnet localhost 9501**

The result will look like this :

```
--------------------------------------------------------------------------------
                    TCServer Telnet Deamon V. 1.0
--------------------------------------------------------------------------------

  Debbugger on Port   : 50682 / Active : 0
  Avg. response Time  : 1216 ms.
  Nb Session in Use   : 0
  Total Transaction   : 498
  Last                : Mon Apr 14 13:19:22 GMT-05:30 2008

--------------------------------------------------------------------------------

1)    stop server          Close all Session and Stop Server.
2)    reset                Stop all Listeners, reload the config, restart
3)    adapter list         Return the List of Adapters
4)    listener list        Return the List of Listeners
5)    stop listener...     Stop a listener
6)    start listener...    Start a listener
7)    reload config        Reload the configuration file tcserver.xml
8)    reset data           Reset the statistices data.
9)    stop adapter ...     Stop connections on GLOBUS.
x)    disconnect           Quit this telnet Session

Please choose a value : [ ]

--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
```

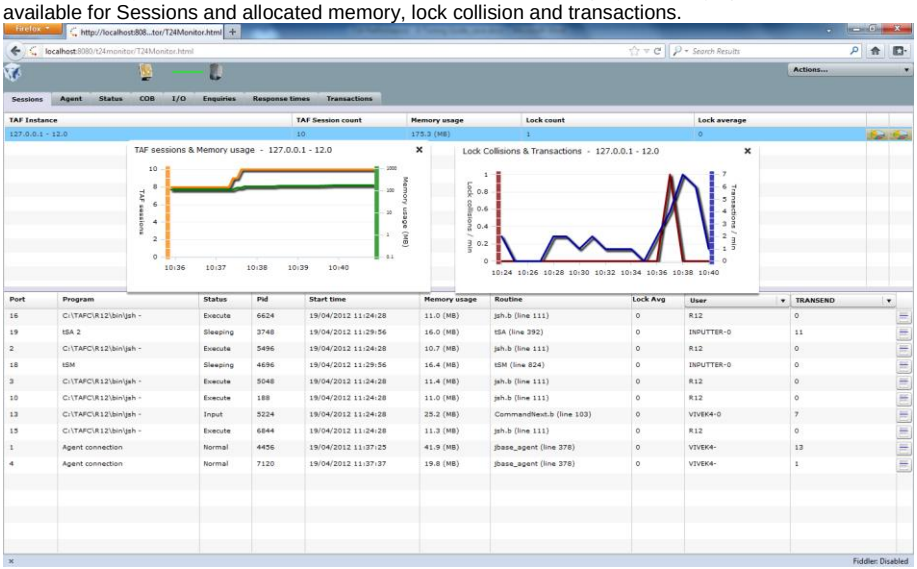
Refer to the TC user guide for more information on monitoring TCServer.

## T24 Monitor

T24Monitor is a web application intended to show real-time statistics and inform about overall health of the T24 system. T24Monitor is the central piece of the monitoring system, not only a web application but also responsible for collecting and storing monitoring events received from remote T24 instances. JMX beans exported by T24Monitor make these statistics available to other applications that are able to access these monitoring beans.

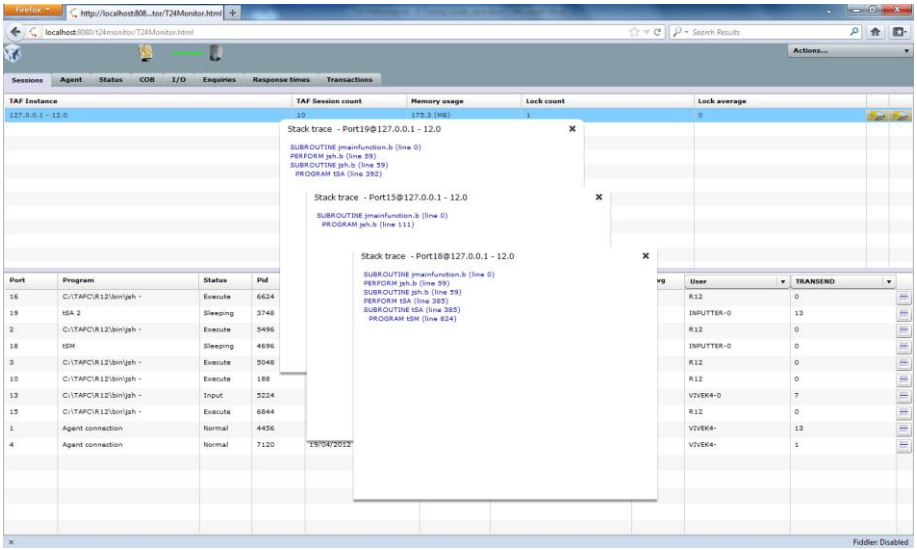
## Sessions

In Sessions tag - TAFC instance and TAFC session details are displayed. The graphical view are also available for Sessions and allocated memory, lock collision and transactions.



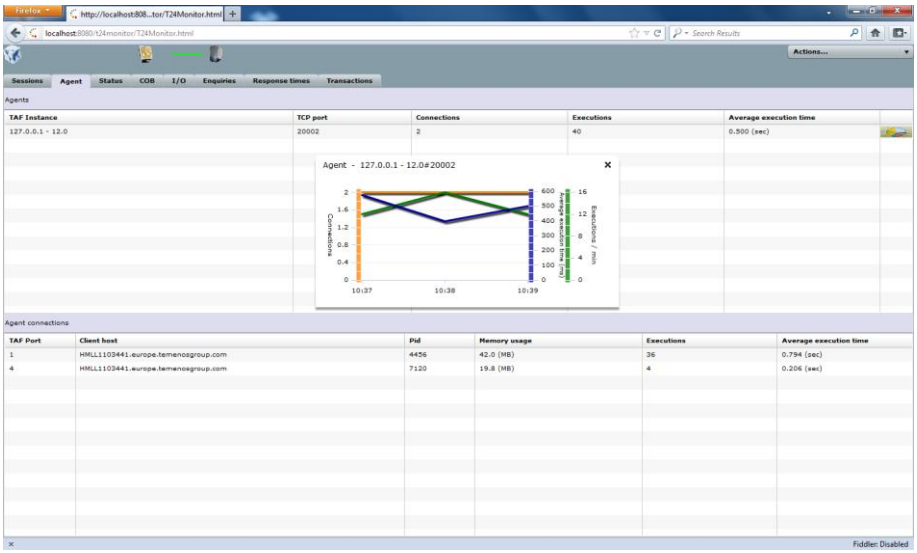

## Sessions stack trace

Sessions stack view is also available for the individual sessions.



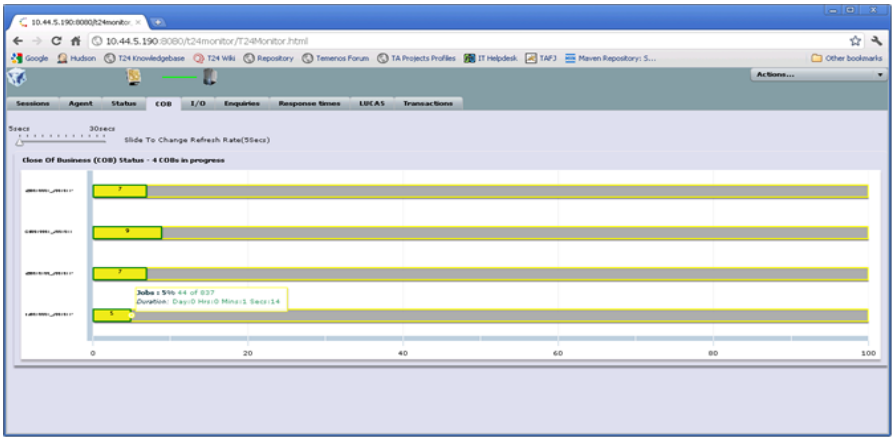
## Agents

Agent wise information like Port, process id, allocated memory, bytes transferred are displayed under this tab.



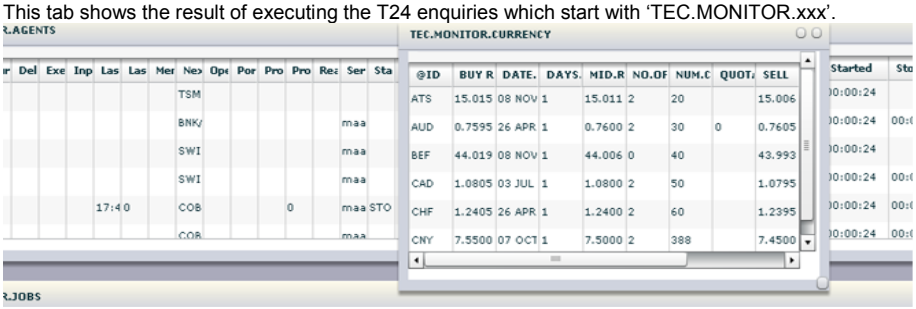

## COB

This tag shows the progress of the COB for the active company. In the graphical view the horizontal bar shows the percentage of the COB jobs remaining to complete.



## ENQUIRIES

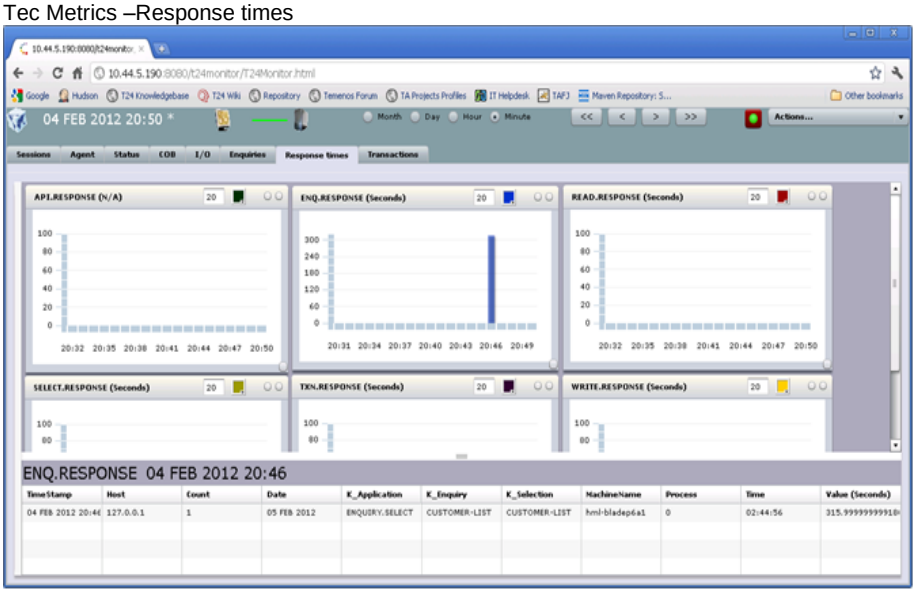
This tab shows the result of executing the T24 enquiries which start with 'TEC.MONITOR.xxx'.



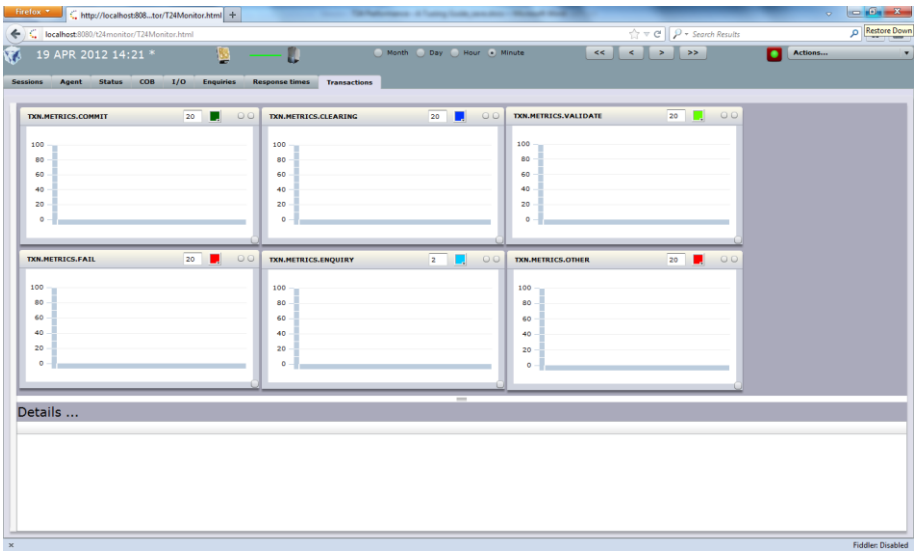

## TEC Metrics

T24 metrics are shown as bar charts in which each bar can be selected. Selecting a particular bar will cause the "Details" table to be populated with the list of TEC events that have contributed to the value of the selected bar.

Tec Metrics –Response times

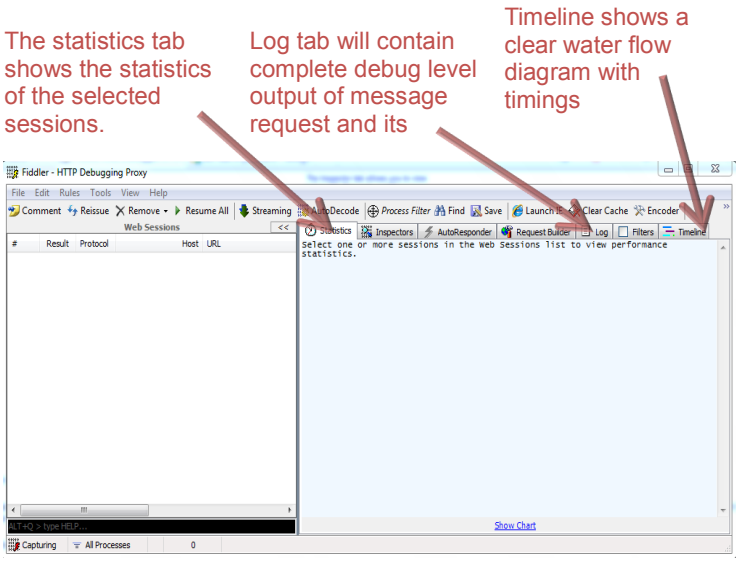


## Tec Metrics –Transactions

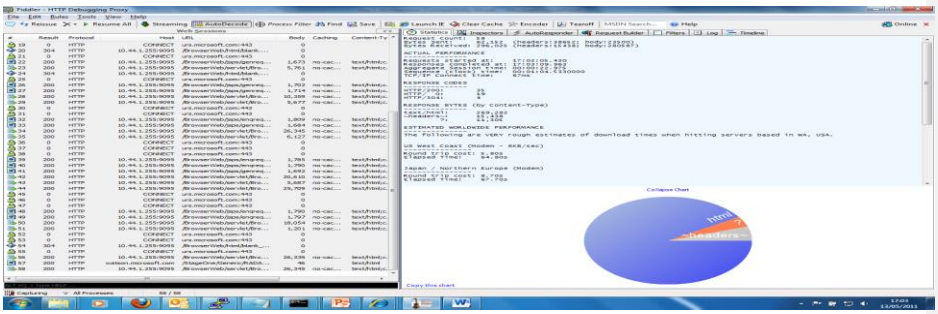


## FIDDLER

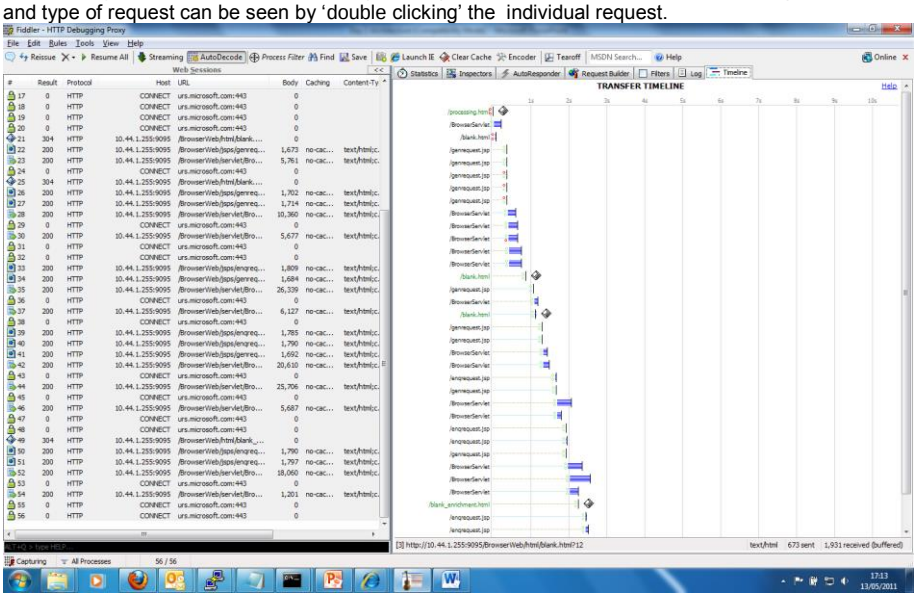
Fiddler is a freeware tool easy to deploy and configure to analyze the statistics of the http requests. It provides details of all the HTTP sessions and also provides the statistics for each of the session request and response and the time spawn. Requests and responses can also be simulated to analyze the web layer connectivity.



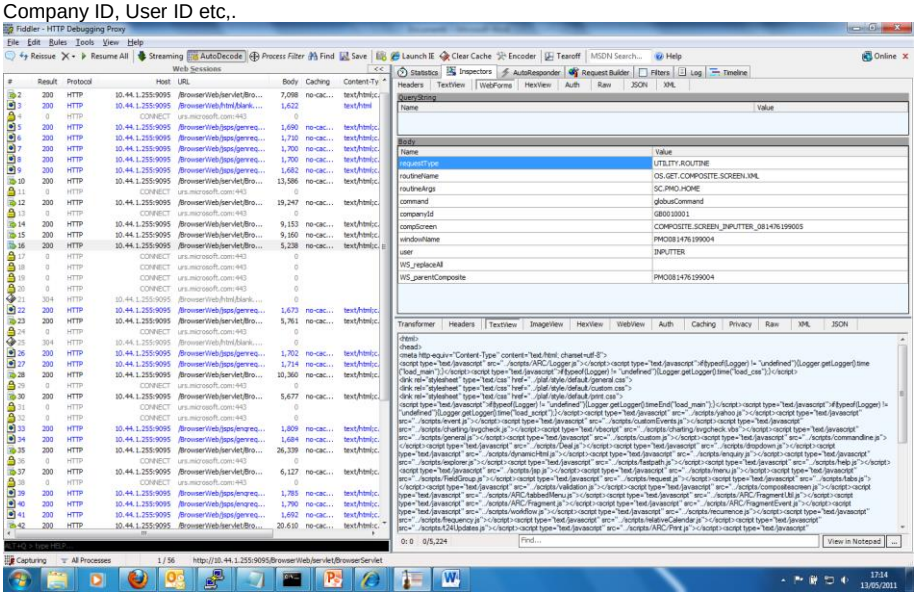

Statistics tab provides the timings for the complete cycle of the transaction request and response which is convenient to analyse the T24 Browser requests to find the function call/component causing the delay.



Timeline tag shows time line for one complete request where in which the individual component time and type of request can be seen by ‘double clicking’ the individual request.



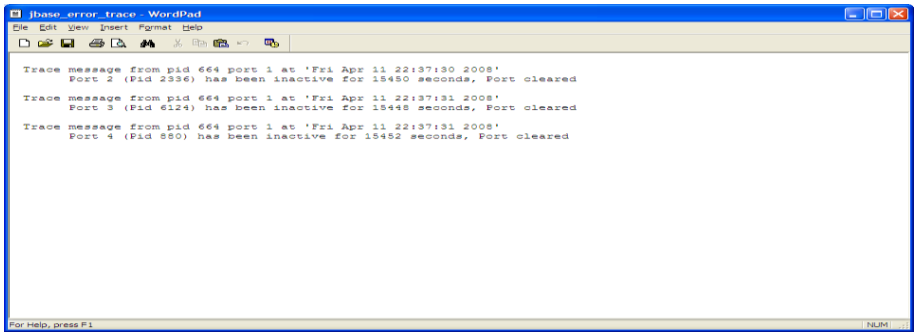

The inspector screen shows the target request details like Routine name, Routine type, Routine args, Company ID, User ID etc,.



jBASE

## Log files

Jbase_error_trace under $JBCRELEASEDIR/tmp directory. This file gives very useful information about jbase, jbase will automatically logs jbase system errors into this file. This file needs to be checked on regular basis to know jbase level issues and solve them.




## WHERE

WHERE command is very useful to analyse / monitor a particular jBASE or T24 process. WHERE gives useful information like what the processes is doing, what routine its executing, what line the process is in, how many reads writes opens deletes it has done etc

```
Port    Device      Account       PID       Command
40      0           cit10         4075912   Port 40 details at 20:13:16 14 APR 2008
                                            Thread type Normal
                                            Usage Count 1M from 1M
                                            Application Open Files 62 , Actual O/S Open Files 57
                                            Memory: Free 1,392,320   Used 4,634,656
                                            READ's 15556 , WRITE's 4603
                                            DELETE's 1 , CLEARFILE's 0
                                            EXECUTE's 4 , INPUT's 0
                                            OPEN's 97
                                       ---- jsh -Jb -c tSA 7
                                            Program performing EXECUTE/PERFORM at jsh.b,59
                                       ---- tSA 7
                                            Program is SLEEPING at BATCH.JOB.CONTROL,545
jsh pzero ~ -->
```

*Screen shot of WHERE (V portnumber output*

## SHOW.ITEM.LOCKS or jrla –dvL or jdls -dvL

Any of above commands can be used to monitor the locks currently held in jbase level based on the type of locking used in the current environment.

If jrla or jdls is used then use the following command to monitor the locks
**Jrla –dvL or jdls –dvL**

```
  PORT       PID FILENAME                                RECORDKEY                                LOCK#          PORT

    69   3289390 ../bnk.data/eb/F.TSA.STA000                                                     0x00002000,R
   106   5804468 ../bnk.data/am/FBNK.AM.DY5072           US-SECURITIES*20070911                  0x4a99d946,W
   107   2523156 ../bnk.data/eb/F.TSA.STA000             10288                                  0x1f2d4e62,W
   107   2523156 ../bnk.data/eb/F.TSA.STA000                                                     0x00002000,R
   129   5431646 ../bnk.data/ac/FBNK_EXT_REPORT#NAU      DBGG3.SM.CUST.M                        0x4ea0fc10,W
   289   5820808 ../bnk.data/de/F_SWIFT_OUT_LIST         D20080408000077646702.4                0x1ea10e12,W
   312   5615634 ../bnk.data/de/F.DE.O.PR014             N                                      0x72b81df1,W
   312   5615634 ../bnk.data/de/F.DE.O.MS015                                                     0x00001000,R
   312   5615634 ../bnk.data/de/F_PRINT_OUT_LIST         D20080408000077646702.3                0x19a10633,W
   312   5615634 ../bnk.data/de/F.DE.O.HE003             D20080408000077646702                  0x40764b08,W
```

## J4 File Size

This is one of main things to monitor when running in J4 (jBASE) as database, file sizing place an important role in getting the performance throughput in jBASE environment.

Plenty of utilities are available in jBASE to monitor and analyse J4 file size, the most used command is

**jrf –R filename**

This will give some indication on file size and what will be best modulo for the current size of the file

```
jsh agntest ~ -->jrf -R FBNK.ACCOUNT
../bnk.data/ac/FBNK.ACCOUNT Type J4, 149 records at size 480, Downsizing skipped from modulo 1877 to 21.
jsh agntest ~ -->jrf -R FGB1.ACCOUNT
../bnk.data/ac/FGB1.ACCO000 Type J4, 249097 records at size 708, Resize from modulo 53249 to 54801.
jsh agntest ~ -->
```

**jstat –v filename**

Gives more useful information about the file and the record size etc

```
jsh agntest ~ -->jstat -v FGB1.ACCOUNT
File ../bnk.data/ac/FGB1.ACCO000
Type=J4 , Hash method = 3
Create at Sun Mar 30 18:49:14 2008
Groups = 53249 , Frame size = 4096 bytes , Secondary Record Size = 8192 bytes
Restore re-size parameters : (none)
File size = 260075520 bytes , Inode = 465773 , Device = Id 2686977
Last Accessed Sat Apr 12 16:02:33 2008 , Last Modified Sat Apr 12 16:02:33 2008
Backup = YES , Log = YES , Rollback = YES , Network = AUTO
Secondary control file = ../bnk.data/ac/FGB1.ACCO000]I


Record Count = 249097 , Record Bytes = 176520104
Bytes/Record = 708 , Bytes/Group = 3314
Primary   file space:   Total Frames = 63400 , Total Bytes = 176203897
Secondary file space:   Total Frames = 94 , Total Bytes = 316207
jsh agntest ~ -->
```

Interpretation of jstat output

- Avoid having huge bucket size
- Re-hash if the secondary size is huge
- Avoid logging for the files not to be replicated to the secondary server while using TJ
- Check whether the secure mode is really needed for the JR files.
- Check the file growth rate and adopt the regular resize policy in place.
- Example to create a file with ‘n’ no of records with appropriate size of each record ‘s’

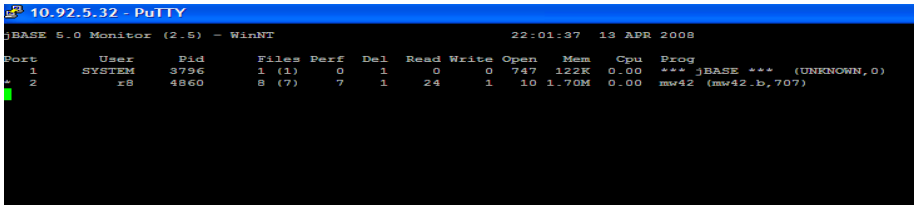
## Mw42 (My WHERE)

mw42

Usage: Options [Sleep (Default 5s)] [LoopCount (Default 7200)]

-a display application account instead of logged on user name
-f display Full listing, normally Command is truncated to screen size
-m split the memory into Free and Used sections
-n disable check for input (some platfoms block when we do this)
-pPort restrict display to Port number
-pp1-p2 restrict display to Port range p1 to p2
-s display IO stats instead of iSeries JobId (Default on UNIX)
-t prefix each line with time stamp
-uuser restrict display to user
You can toggle most of the options at runtime by hitting the key for the option

q or 'Q' will exit




## Operating System

OS tools are a valuable tool in determining the type of limitation that is being applied, and from there

### AIX

The main tool for AIX is nmon. Before a test is run the following command should be run:

nmon -r <<testname>> -t -F <<filename>> -s<<timeslice>> -c<<duration>>

e.g.

nmon -r eod1 -t -F /tmp/eod1_nmon_file -s30 -c300

The above command would create a file in /tmp call eod1_nmon_file and would give the test a title of eod1. This command would record stats every 30 seconds for 300 minutes. Once the test is complete (you do not have to wait for the time period to finish) the output needs sorting. Still in AIX use the following command:

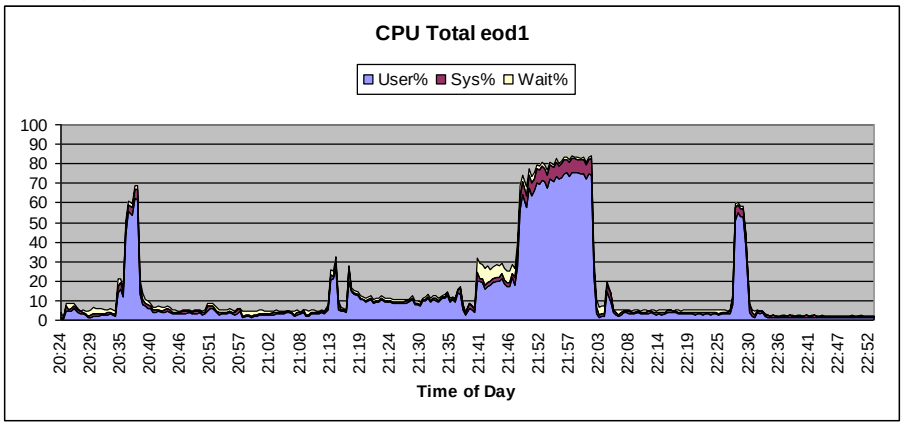
Sort –A infile > outfile

e.g. sort –A /tmp/eod1_nmon_file > /tmp/eod1_nmon_file.csv

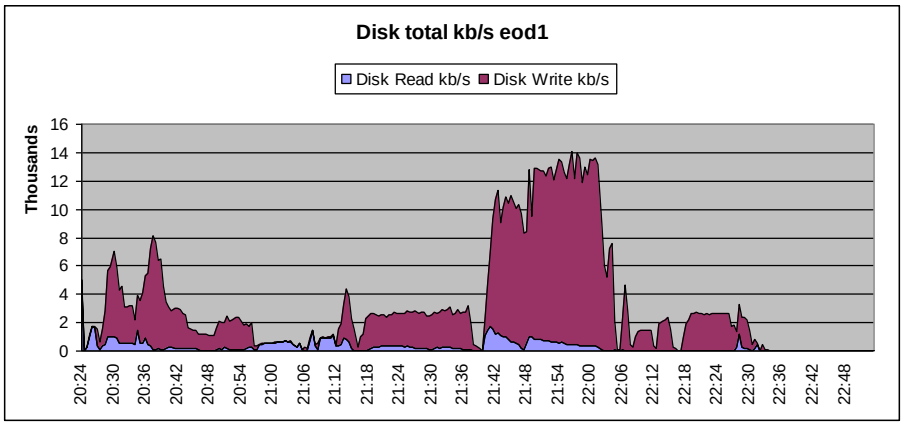
This command sorts the data into a .csv file. This file should then be ftp'd to a windows machine, and the "nmon analyzer" spreadsheet used to analyse the data.

NB the “nmon analyzer” is an IBM tool, though it is unsupported. The T & R department can supply copies of this spreadsheet if required.

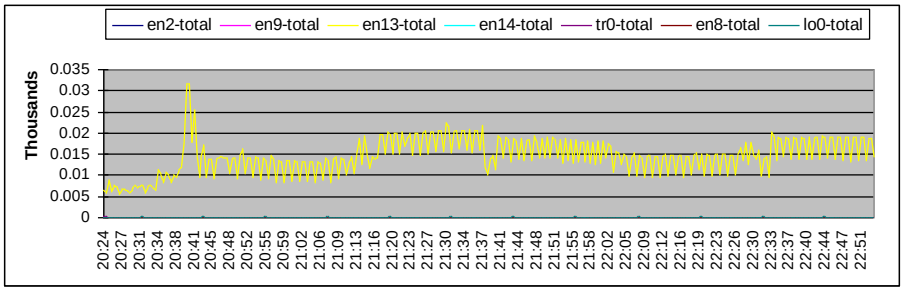
Below are some examples of the data that is presented by this tool:
CPU usage:



Disk activity:




Network utilisation:

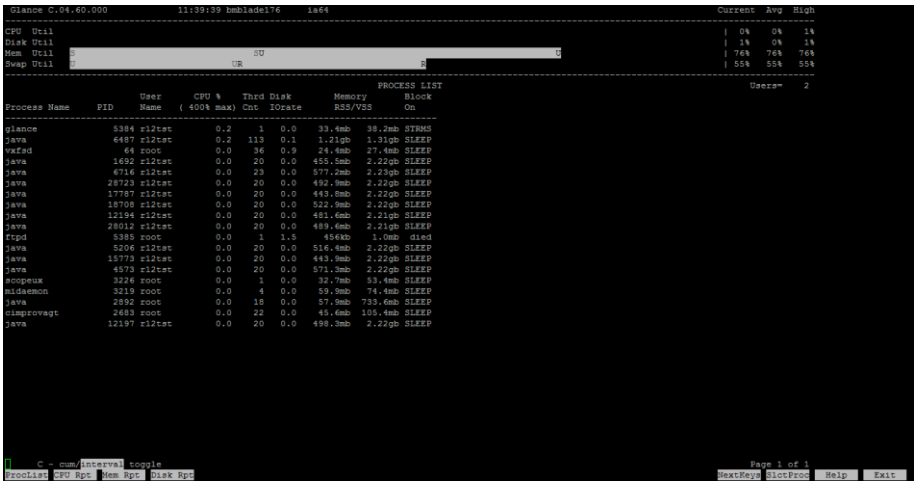



IBM default OS parameters done in BM

## HP – tba

### Glance plus /top

GlancePlus is a powerful and easy-to-use online performance diagnostic tool for HP-UX systems. It is distributed in two forms: "gpm", a Motif-based program and “glance", a character mode program. We provide both so you can choose the most appropriate tool for your job. With gpm, you get the power and ease-of-use of a Motif-based tool. With glance, you can run on almost any terminal or workstation, over a serial interface and relatively slow data communication links, and with lower resource requirements. Either component provides the same rich set of performance information. The default Process List screen provides general data on system resources and active processes. More specific data is presented via the CPU, Memory, Disk IO, Network, NFS, System Calls, Swap, and System Table screens. Specific per-process detail is also available via the individual process screens. Running in a terminal environment, Glance aids performance problem resolution on all HP 9000 series systems.



Command options for glance

| Command | Screen Displayed / Description |
|---|---|
| a | CPU By Processor |
| c | CPU Report |
| d | Disk Report |
| g | Process List |
| i | IO By File System |
| l | Network By Interface |
| m | Memory Report |
| n | NFS By System |
| t | System Tables Report |
| u | IO By Disk |
| v | IO By Logical Volume |
| w | Swap Space |
| A | Application List |
| B | Global Waits |
| D | DCE Global Activity |
| G | Process Threads |
| H | Alarm History |
| I | Thread Resource |
| J | Thread Wait |
| K | DCE Process List |
| N | NFS Global Activity |
| P | PRM Group List |
| T | Transaction Tracker |
| Y | Global System Calls |
| Z | Global Threads |
| U | IO By HBA Card |
| V | Logical System List |
| K | Logical System Report |
| ? | Commands Menu |
| S | Select a NFS system/Disk/Application/Trans/Thread |
| s | Select a single process |
| F | Process Open Files |
| L | Process System Calls |
| M | Process Memory Regions |
| R | Process Resources |
| W | Process Wait States |

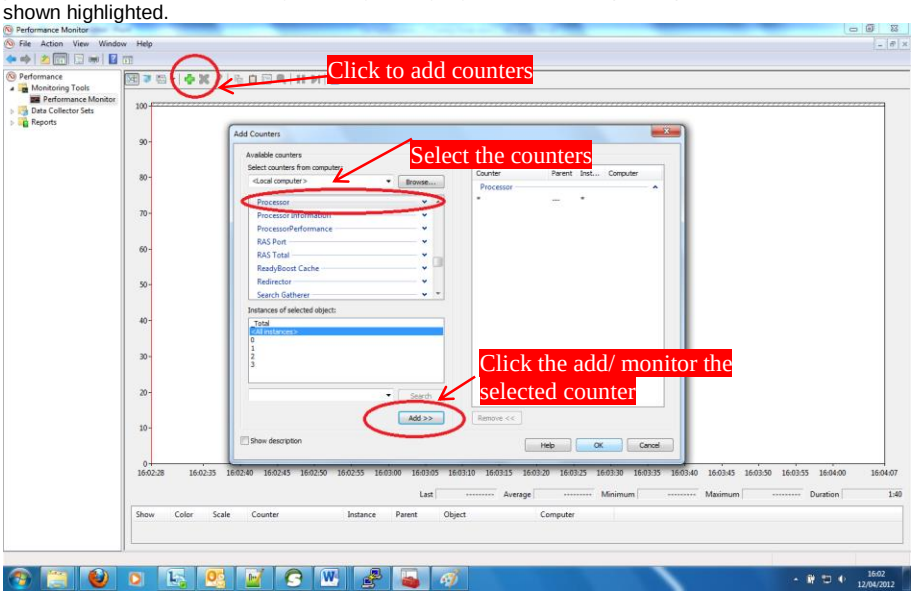
## Windows

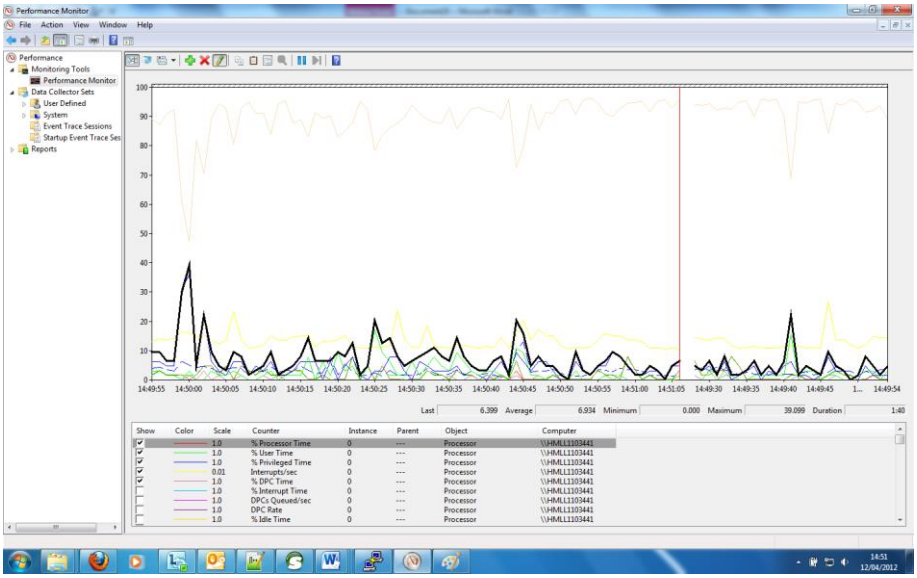
### Perfmon

Microsoft Performance monitor is a simple yet powerful visualisation tool for viewing performance data, both in real time and in log files. In performance monitor, it is possible to load the required performance counters to analyse the system resource (CPU, Memnory, Disk, Network) individually.
Performance monitor aka 'Perfmon' can be started from the Start menu (Click Start, type Perfmon, and then click Perfmon.exe).

## CPU

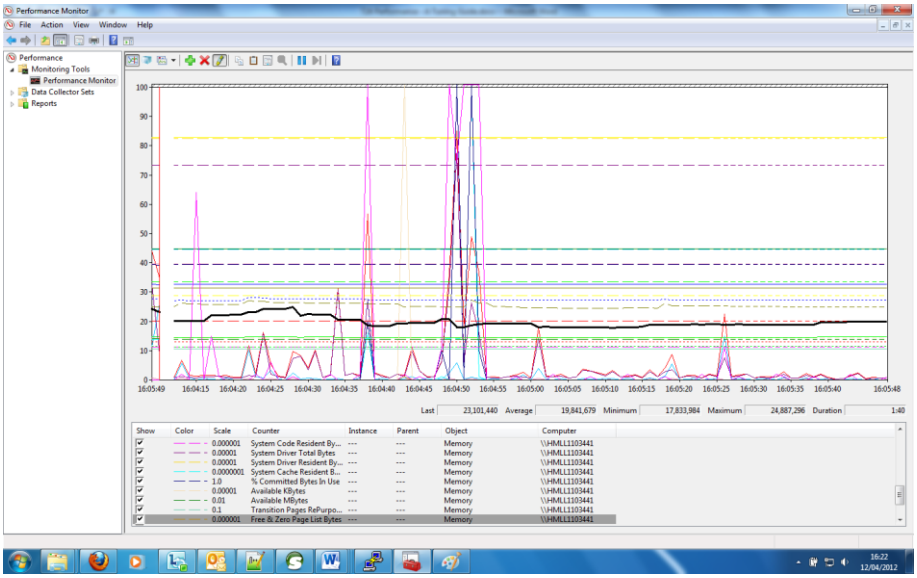
To monitor only the CPU activity, remove all the performance counters and load only the 'processor' performance counters and navigate using the highlighter to see the specific performance counter shown highlighted.



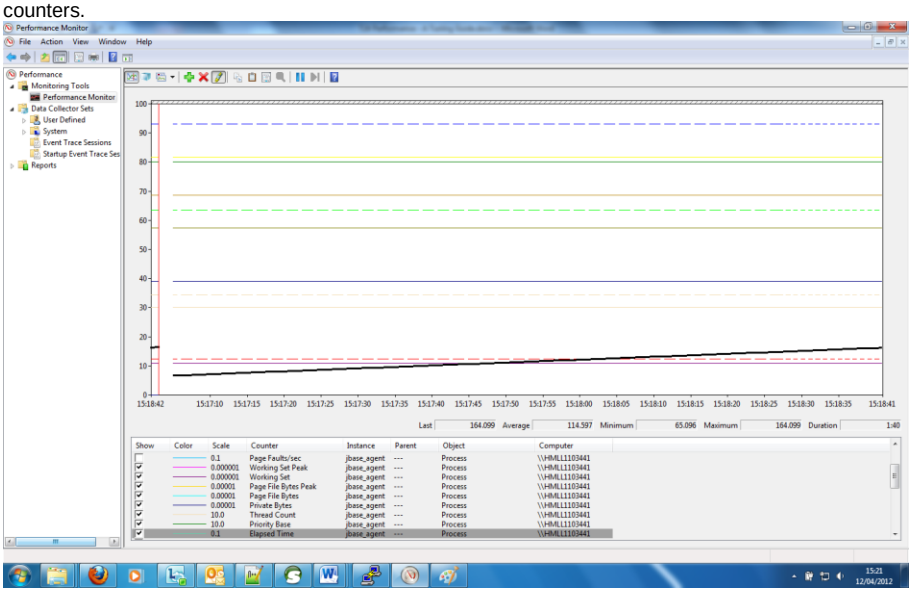
## Memory

Load the memory related performance counters to monitor the memory usage and statistics.



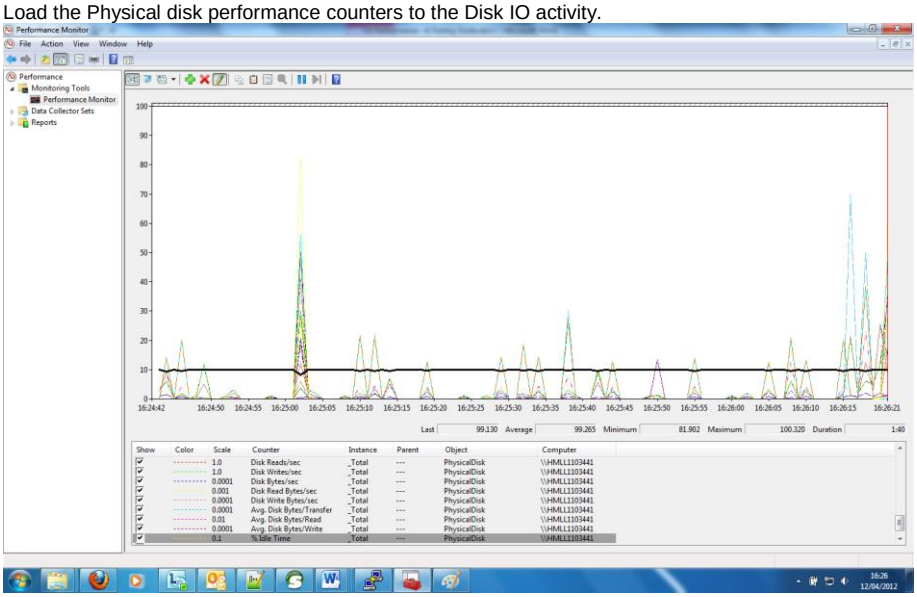
## Process

To monitor a particular process performance load the process and monitor the required performance counters.



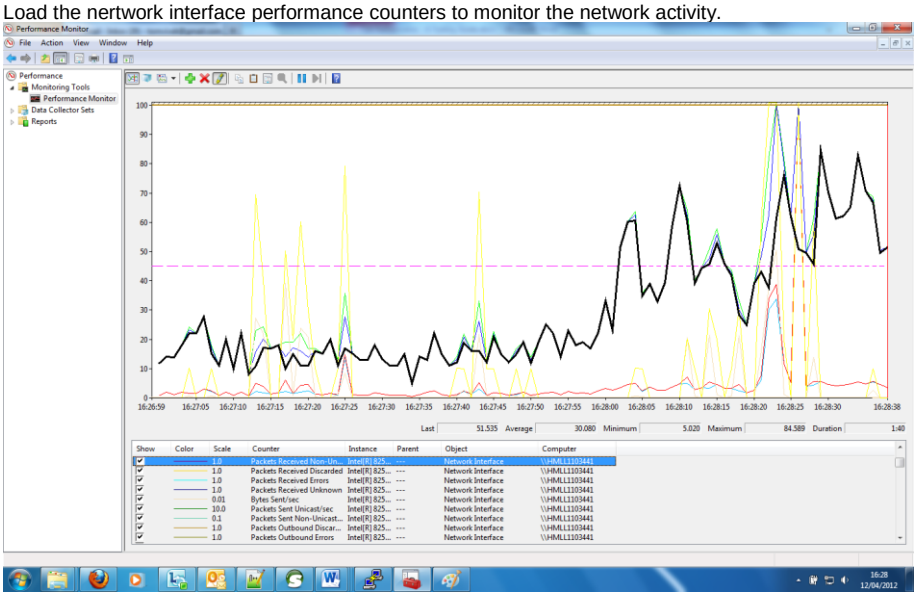
## Disk

Load the Physical disk performance counters to the Disk IO activity.



Network

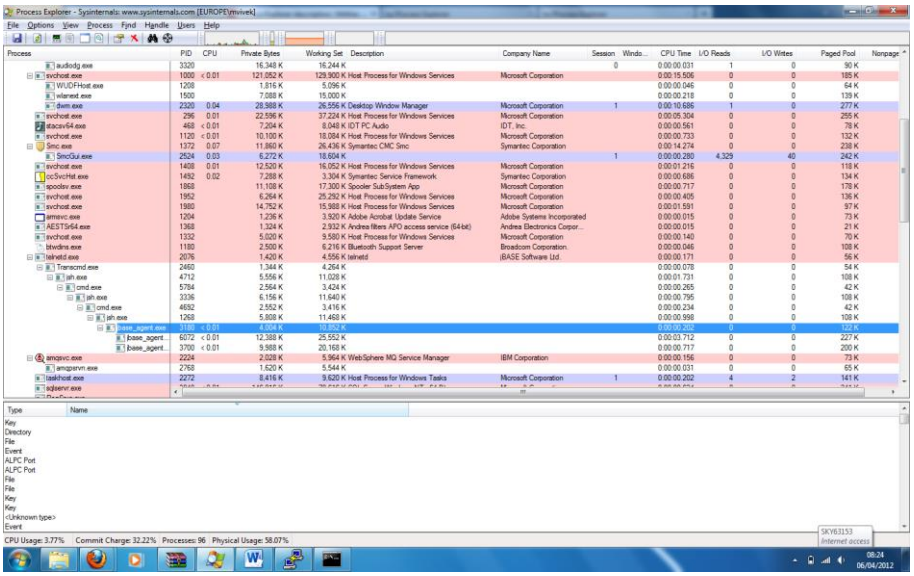
Load the nertwork interface performance counters to monitor the network activity.



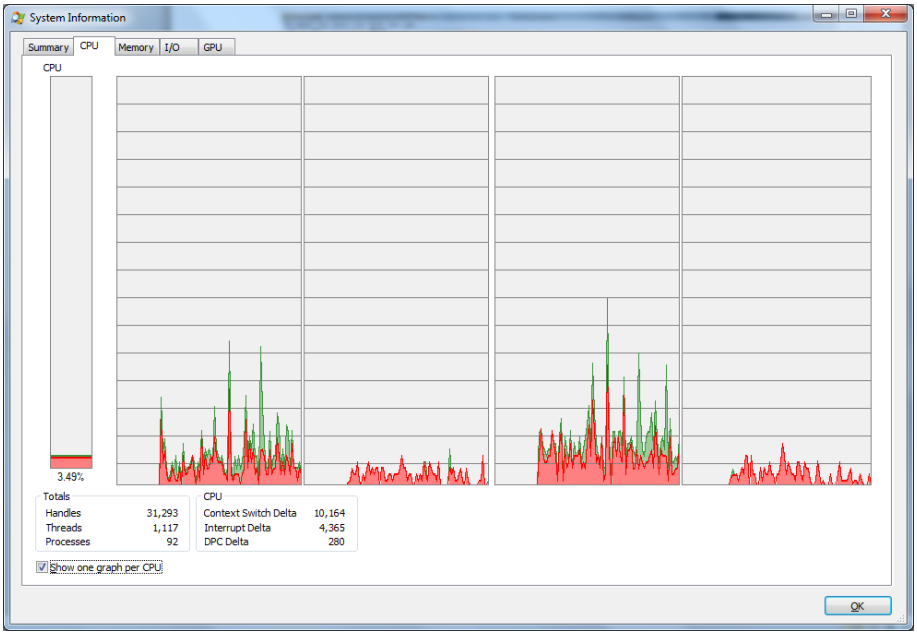
## Process explorer

Process explorer is used to monitor the currently active processes in the system. The Process Explorer display consists of two sub-windows. The top window always shows a list of the currently active processes, including the names of their owning accounts, whereas the information displayed in the bottom window depends on the mode that Process Explorer is in: if it is in handle mode you'll see the handles that the process selected in the top window has opened; if Process Explorer is in DLL mode you'll see the DLLs and memory-mapped files that the process has loaded

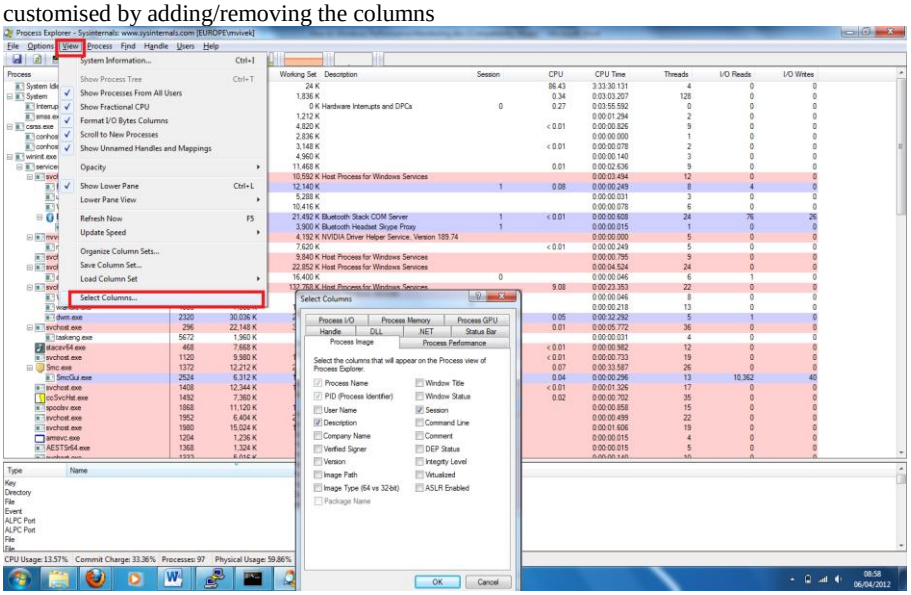
To launch process explorer download ‘process explorer’ and double click the ‘procexp.exe’.



Process explorer provides graphical view of the system performance under different tags as below:



In Process explorer to get more details about processes, more views are available and can be customised by adding/removing the columns



## Resource Monitor

**Formatted:** Heading 4

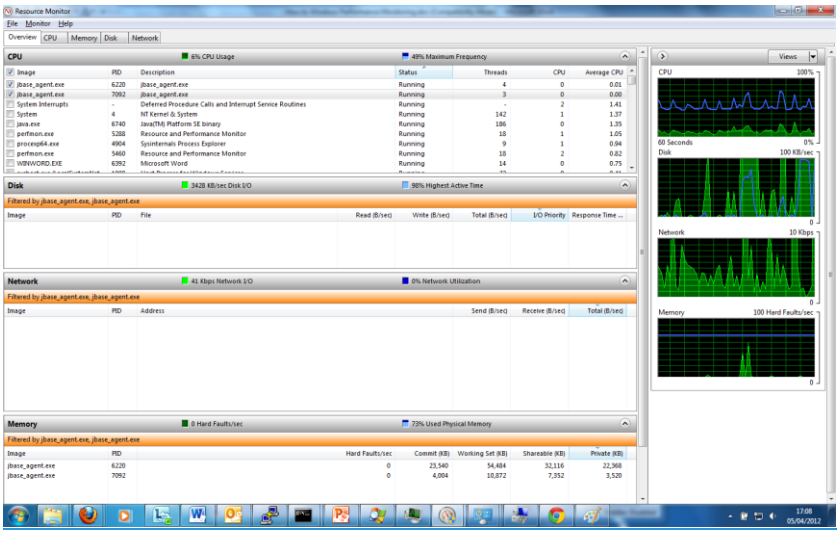
**Resource Monitor** is used to view information about the use of hardware (CPU, memory, disk, and network) and software (file handles and modules) resources in real time. Resource Monitor can be launched:

1. by executing *resmon.exe* through the START menu.
2. through the Task Manager – At the bottom of the task manager by clicking 'Resource Monitor' tab

**Formatted:** Indent: Left: 1.27 cm, No bullets or numbering

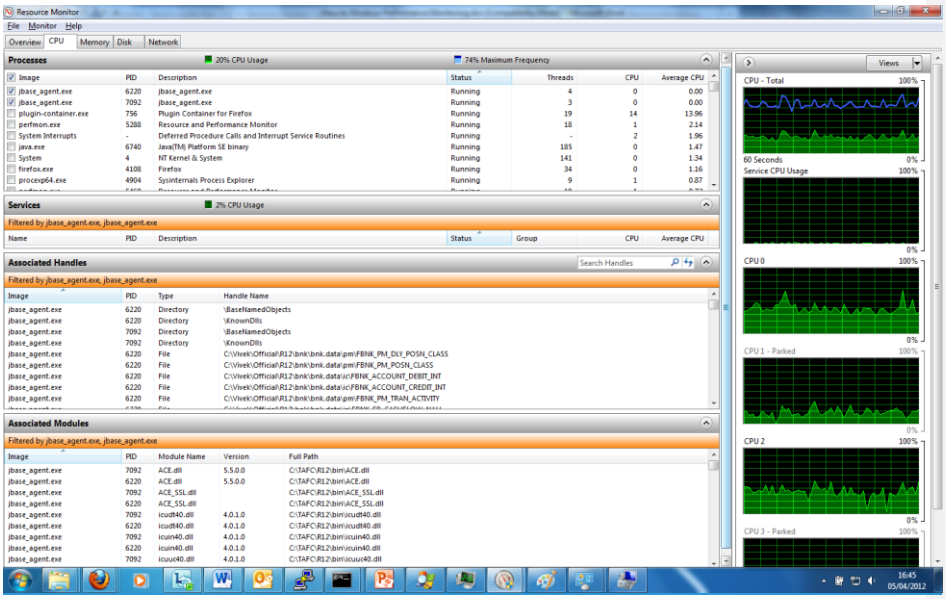
Resource Monitor appears as follows – by default with the Overview screen. The four main performance counters' screen appears on the different menu tags.

Formatted: No bullets or numbering



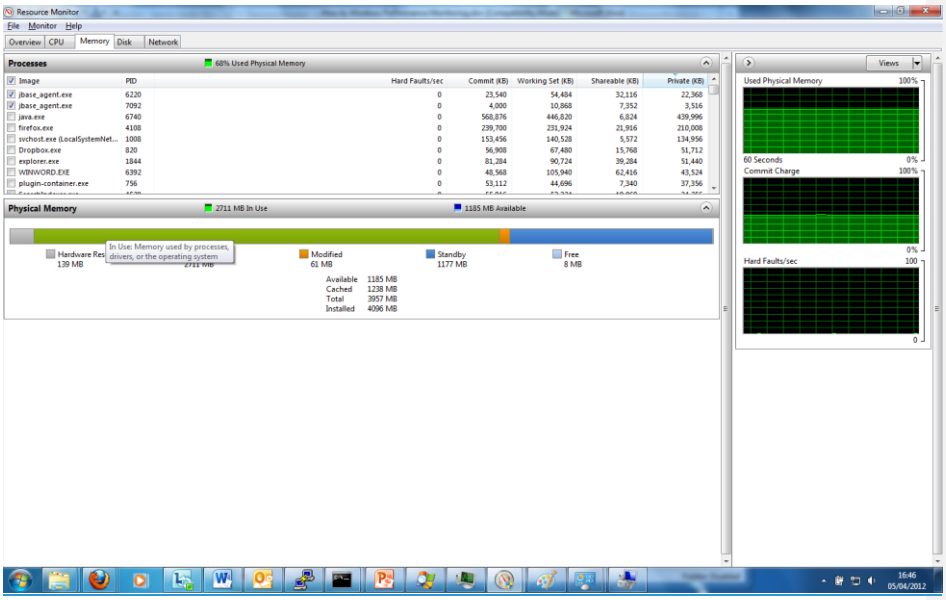
To analyse which processes are causing more CPU click the CPU tag

Formatted: No bullets or numbering



Click the Memory tab to see how your computer's memory is being used.

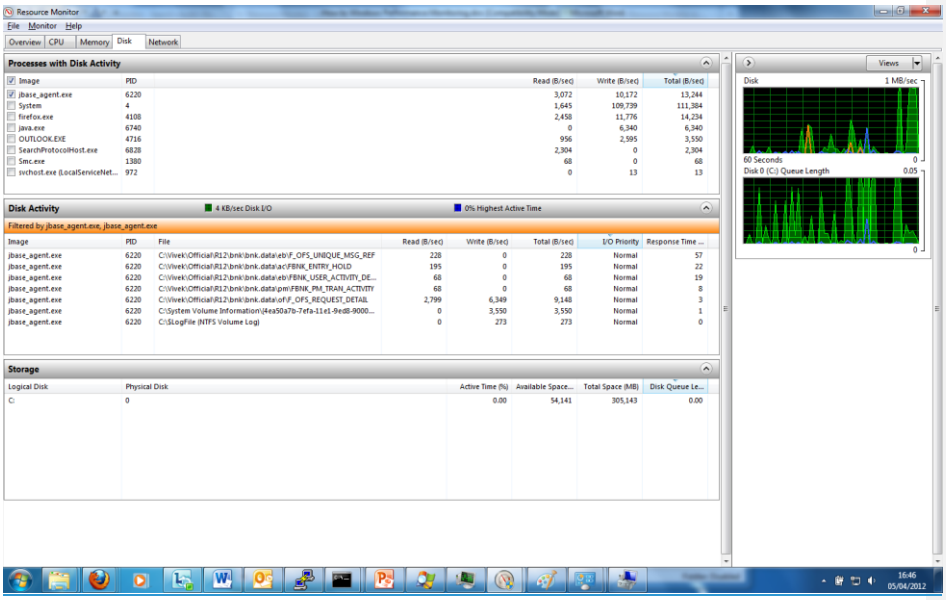
Formatted: No bullets or numbering




The memory usage in total for the whole system (usage split by 'Hardwared reserved', 'In Use', 'Modified', 'Standby' and 'Free' Momory ) is shown in the bottom of the screen.

Formatted: No bullets or numbering

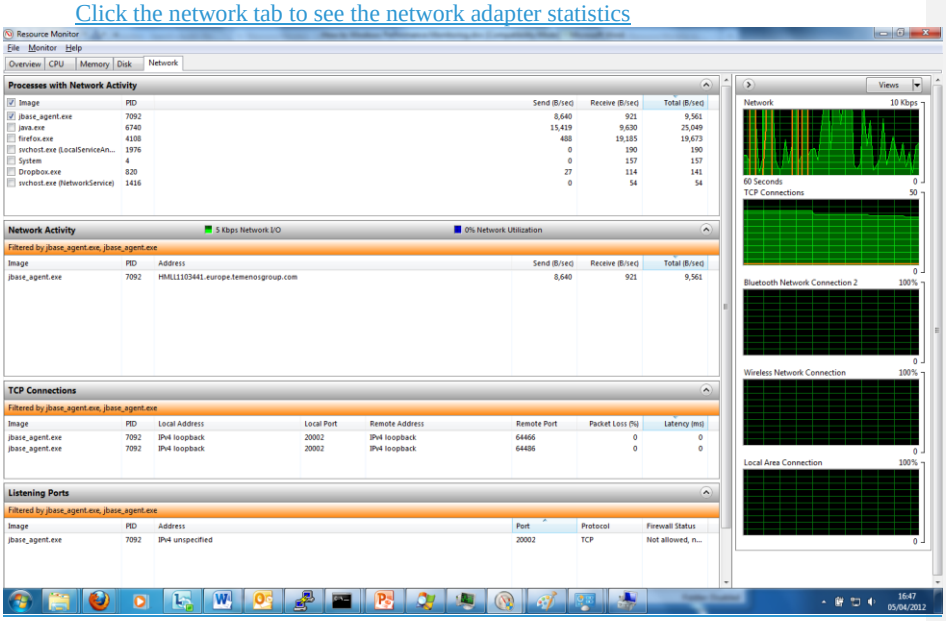
Click the Disk tag to see the disk IO (read & write) for the process.




The Disk Queue lenth can be seen by storage wise – where if the higher value indicates the possible issue in Disk IO.

**Formatted:** Space Before: 0 pt, After: 0 pt, No bullets or numbering

Click the network tab to see the network adapter statistics




Default WIN kernal parameter to check with Dammika

Formatted: Heading 4

**For SQL check the sql profiler.**

## T24 Server

### SEAT

T24 SEAT (System Engineering Automated Testing) performs a trace analysis to record I/O, calls made, path length, warnings etc. for every transaction (online & COB).
The trace analysis provides us with a mechanism to monitor the likely performance and scalability of T24 transactions without having to invest in heavily populated databases. By analysing the I/O and lock profile you can easily determine whether a transaction will scale.
~~Note: The tracing mechanism only works with jBASE 4.1 Major 4.1 , Minor 4 , Patch 0245 (Change 22826) or higher.~~ SEAT would be ideal to monitor/analyse T24 IO foot print issues

## The Mechanics

To turn on the SEAT go to the SPF and set the OP.CONSOLE field to ON. When this is set every online (browser, desktop, classic, ofs etc.) transaction ~~and job @RecordRoutine~~ will be traced ~~during COB~~ and their I/O etc recorded in file SEAT.RESULTS.

To switch on SEAT in the COB, Set the OP.CONSOLE field to ON in SPF and create a record in F.LOCKING with ID as SEAT.TRACE. In the record created include the JOB as 'Processname–JobName' shown below:

```
*File F.LOCKING , Record 'SEAT.TRACE'
Command-> 
0001 BNK/AA.EOD.PROCESS-AA.SERVICE.PROCESS
----------------------------------------
```

Formatted: Font: Calibri

When COB is executed with SEAT on, SEAT creates a record for every job in the SEAT.RESULTS table and every record created follows the following syntax:

Batch Record Id*Job Name*Actual Id that has been processed*Date of processing

## Steps involved in Capturing SEAT

These are the following steps involved in capturing SEAT.

### Capture SEAT Info

a. Clear the file F.SEAT.RESULTS for any previous results summary from jBASE Shell.
   Cmd: CLEAR.FILE F.SEAT.RESULTS
b. Change File SPF record SYSTEM field OP.CONSOLE to ON

```
@ID................ SYSTEM
SYSTEM.SPEC........ SYSTEM
SITE.NAME.......... EBS LONDON.
OP.MODE............ O
OP.CONSOLE......... ON
MAIN.ACCOUNT....... ../bnk.data
```

c. Logout and Login into jbase.

d. Input the Required Transaction for which SEAT results needs to be captured. Transactions can be inputted thru any interface like Browser, Classic, OFS or Desktop. For Example in order to capture the seat for Funds Transfer, the following is required

- ✓ Form a FT OFS string and copy it to clipboard.

  FUNDS.TRANSFER,TEST/I/PROCESS,INPUTT/123456,/,TRANSACTION.TYPE::=AC,DEBIT.ACCT.NO::=28703,DEBIT.CURRENCY::=USD,DEBIT.AMOUNT::=225T,CREDI
  T.ACCT.NO::=16985

- ✓ Run tSS <Ofs source>from jshell

- ✓ Paste the OFS string and hit Enter.

```
jsh pzero ~ -->tSS GCS
<tSS version="1.1"><t24version>200712.001</t24version><t24pid>5530000</t24pid><t24ofssource>GCS</t24of
ssource><clientIP/></tSS>
FUNDS.TRANSFER,TEST/I/PROCESS,INPUTT/123456,/,TRANSACTION.TYPE::=AC,DEBIT.ACCT.NO::=28703,DEBIT.CURREN
CY::=USD,DEBIT.AMOUNT::=225T,CREDIT.ACCT.NO::=16985
```

- ✓ The output will be something as follows

```
FT0724300233//1,TRANSACTION.TYPE:1:1=AC,DEBIT.ACCT.NO:1:1=28703,CURRENCY.MKT.DR:1:1=1,DEBIT.CURRENCY:1
:1=USD,DEBIT.AMOUNT:1:1=225000.00,DEBIT.VALUE.DATE:1:1=20070831,CREDIT.ACCT.NO:1:1=16985,CURRENCY.MKT.
CR:1:1=1,CREDIT.CURRENCY:1:1=USD,CREDIT.VALUE.DATE:1:1=20070831,PROCESSING.DATE:1:1=20070831,CHARGE.CO
M.DISPLAY:1:1=NO,COMMISSION.CODE:1:1=DEBIT PLUS CHARGES,CHARGE.CODE:1:1=DEBIT PLUS CHARGES,PROFIT.CENT
RE.CUST:1:1=100130,RETURN.TO.DEPT:1:1=NO,FED.FUNDS:1:1=NO,POSITION.TYPE:1:1=TR,AMOUNT.DEBITED:1:1=USD2
25000.00,AMOUNT.CREDITED:1:1=USD225000.00,DELIVERY.OUTREF:1:1=D2008041400105508080 3-900.1.1       DEBI
T ADVICE,DELIVERY.OUTREF:2:1=D20080414001055080804-910.2.1       CREDIT ADVICE,CREDIT.COMP.CODE:1:1=GB
0010001,DEBIT.COMP.CODE:1:1=GB0010001,LOC.AMT.DEBITED:1:1=225000.00,LOC.AMT.CREDITED:1:1=225000.00,CUS
T.GROUP.LEVEL:1:1=99,DEBIT.CUSTOMER:1:1=100130,CREDIT.CUSTOMER:1:1=100352,DR.ADVICE.REQD.Y.N:1:1=Y,CR.
ADVICE.REQD.Y.N:1:1=Y,CHARGED.CUSTOMER:1:1=100352,TOT.REC.COMM:1:1=0,TOT.REC.COMM.LCL:1:1=0,TOT.REC.CH
G:1:1=0,TOT.REC.CHG.LCL:1:1=0,RATE.FIXING:1:1=NO,TOT.REC.CHG.CRCCY:1:1=0,TOT.SND.CHG.CRCCY:1:1=0,AUTH.
DATE:1:1=20070831,STMT.NOS:1:1=147150010550808.02,STMT.NOS:2:1=1-2,CURR.NO:1:1=1,INPUTTER:1:1=105_INPU
TTER___OFS_GCS,DATE.TIME:1:1=0804141406,AUTHORISER:1:1=105_INPUTTER_OFS_GCS,CO.CODE:1:1=GB0010001,DEPT
.CODE:1:1=1
```

- ✓ Exit from tSS.

e. Change File SPF record SYSTEM field OP.CONSOLE to null.
This is to stop further capturing of transactions

f. Count F.SEAT.RESULTS.

There should be one F.SEAT.RESULTS record.

**Note:-** SEAT must be turned ON only when particular transaction(s) performance needs to be monitored, enabling SEAT will drastically affect performance so this needs extra care.

### TEC.~~MONITOR~~ Temenos Enterprise Console

TEC is based upon a lightweight architecture with minimal impact to T24 installation. This allows users to monitor the behavior of T24 application from the time a user sign on until he signs off and monitors activities while COB and services are running.

TEC has the following TEC.ITEMS available for monitoring:

| TEC.ITEM | Description | Default Threshold Values |
|---|---|---|
| API.RESPONSE | Response time for API call | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 300<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 400<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 600<br>THRESHOLD.TYPE – CRITICAL |
| ENQ.RESPONSE | Enquiry build response time | THRESHOLD.TYPE – INFO<br>THRESHOLD – 1000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 3000<br>THRESHOLD.TYPE – CRITICAL |
| FILE.OPEN | Count of files open | THRESHOLD.TYPE – WARNING<br>THRESHOLD – 1<br>THRESHOLD.TYPE – CRITICAL |
| FILE.SIZE | Size of opened files | |
| LOCK.COLLISION | Count of lock collisions | THRESHOLD.TYPE – INFO<br>THRESHOLD – 1<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 2<br>THRESHOLD.TYPE – CRITICAL |
| READ.RESPONSE | Response times for record read | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 300<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 400<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 500 |

Formatted: No bullets or numbering
Formatted: English (United States)
Formatted: Normal
Formatted: Font: (Default) +Body (Calibri), 12 pt
Formatted Table
Formatted: Font: (Default) +Body (Calibri), 12 pt
Formatted: Font: (Default) +Body (Calibri), 12 pt
Formatted: Font: (Default) +Body (Calibri), 12 pt
Formatted: Font: (Default) +Body (Calibri), 12 pt
Formatted: Font: (Default) +Body (Calibri), 12 pt

| | | |
|---|---|---|
| | | THRESHOLD.TYPE – CRITICAL<br>THRESHOLD – 600 |
| READ.SIZE | Size of read records that have been read | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 4097<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 6000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 8000<br>THRESHOLD.TYPE – CRITICAL<br>THRESHOLD – 10000 |
| SELECT.RESPONSE | Select response time | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 300<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 400<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 500<br>THRESHOLD.TYPE – CRITICAL |
| TXN.METRICS.COMMIT | Total response time for a transaction commit. | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 3000<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 4000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 5000<br>THRESHOLD.TYPE – CRITICAL<br>THRESHOLD – 6000 |
| TXN.METRICS.SERVICE | Throughput metrics for a given service (either ONLINE / COB) showing throughput per minute | |
| TXN.METRICS.OTHER | Total response time for any other OFS requests | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 50<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 100<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 200<br>THRESHOLD.TYPE – CRITICAL |

| | | |
|---|---|---|
| TXN.RESPONSE | Transaction commit response time i.e., from commit processing until database update. Does not include time spent in OFS | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 3000<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 4000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 5000<br>THRESHOLD.TYPE – CRITICAL<br>THRESHOLD – 6000 |
| TXN.METRICS.BROWSER | Response time for utility requests made to T24 to service browser requests. Eg., Menu, enquiry selection screens etc., | |
| TXN.METRICS.CLEARING | Response time for clearing messages | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 3000<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 4000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 5000<br>THRESHOLD.TYPE – CRITICAL<br>THRESHOLD - 6000 |
| TXN.METRICS.ENQUIRY | Total response time for an enquiry request | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 3000<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 4000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 5000<br>THRESHOLD.TYPE – CRITICAL<br>THRESHOLD – 6000 |
| TXN.METRICS.VALIDATE | Total response time for a transaction validate request | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 3000<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 4000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 5000<br>THRESHOLD.TYPE – CRITICAL<br>THRESHOLD - 6000 |

| | | |
|---|---|---|
| WRITE.RESPONSE | Response time for record write | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 300<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 400<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 500<br>THRESHOLD.TYPE – CRITICAL |
| WRITE.SIZE | Size of written records | THRESHOLD.TYPE – IGNORE<br>THRESHOLD – 4097<br>THRESHOLD.TYPE – INFO<br>THRESHOLD – 6000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD – 8000<br>THRESHOLD.TYPE – CRITICAL |
| EXTERNAL.ACTIVITY.STATUS | Tec External Item | |
| CRITICAL.UPDATE | Critical T24 Application Updates | |
| CRITICAL.OVERRIDE | Critical T24 Overrides | |
| CACHE.HIT | Cache hit information | THRESHOLD.TYPE – WARNING<br>THRESHOLD - 5<br>THRESHOLD.TYPE – INFO |
| CACHE.MISS | Cache miss information | THRESHOLD.TYPE – INFO<br>THRESHOLD - 10<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD - 100<br>THRESHOLD.TYPE – CRITICAL |

| | | |
|---|---|---|
| CACHE.STATS | Cache Statistics | THRESHOLD.TYPE – INFO<br>THRESHOLD - 10<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD - 100<br>THRESHOLD.TYPE – CRITICAL |
| TXN.METRICS.FRAMES.ACCESSED | | |
| TXN.METRICS.FAIL | | THRESHOLD.TYPE – IIGNORE<br>THRESHOLD - 5000<br>THRESHOLD.TYPE – INFO<br>THRESHOLD - 7000<br>THRESHOLD.TYPE – WARNING<br>THRESHOLD - 8000<br>THRESHOLD.TYPE – CRITICAL |

Formatted: Font: (Default) +Body (Calibri)

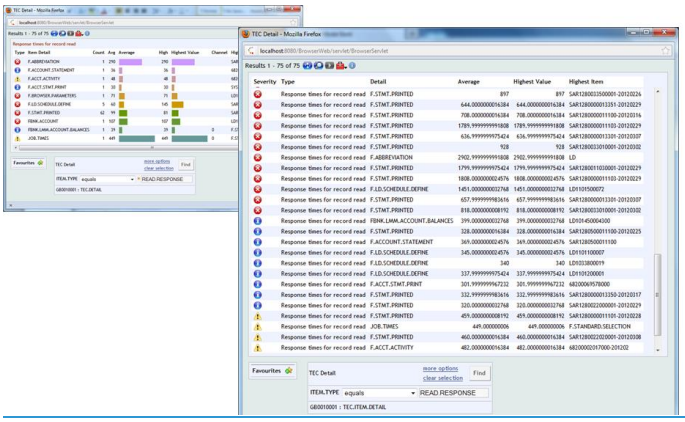
Formatted: Normal

The recorded activities as in the TEC.PROFILE when enabled are written into TEC.OUTPUT file. The ID of the file is <SessionNumber>.<LogSetNumber>.<ItemName>

The consolidated per hour data is updated into the TEC.OUTPUT.HISTORY file. The file contains the Total metric count, metric total and the Top key for every hour since the start of the TEC profiling.

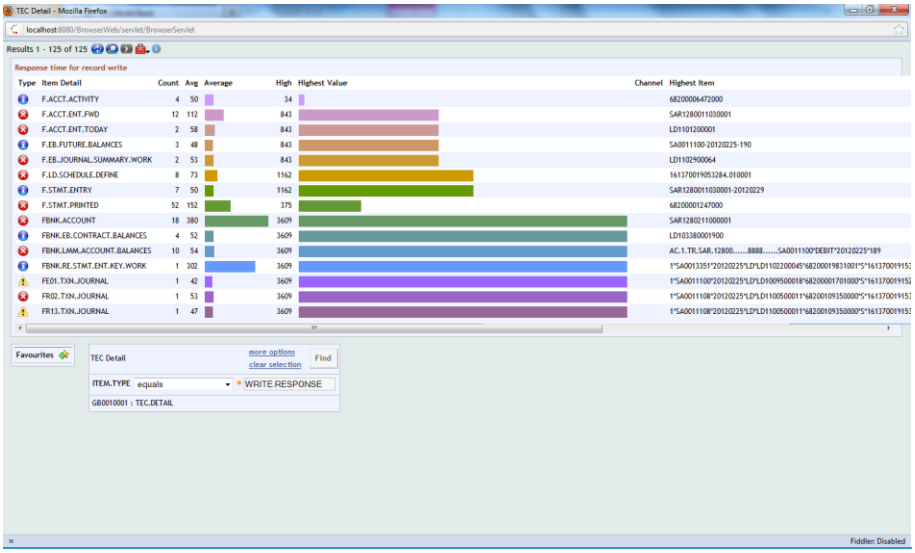
Formatted: Normal

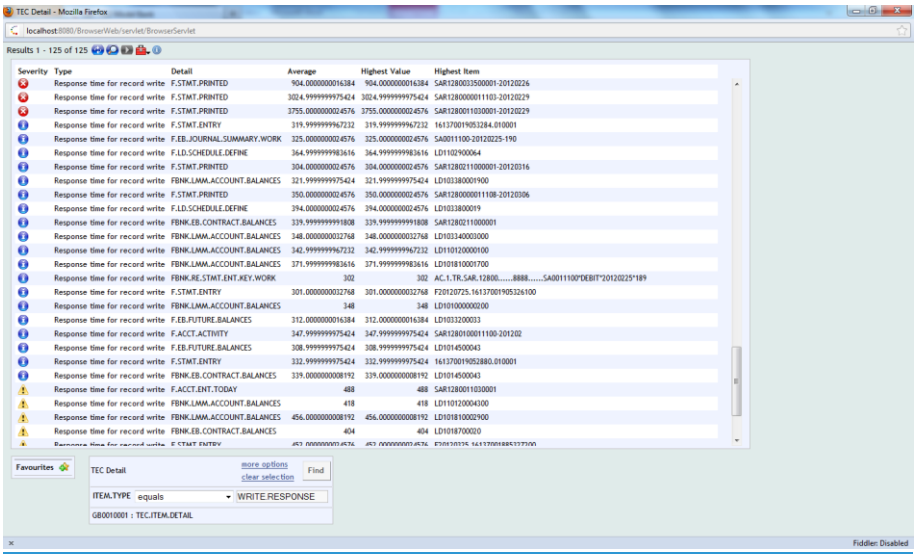
The following example shows the display for a sample recorded activity through the TEC.ITEM & TEC.ITEM.DETAIL enquiries:



For the above it is clear on the record for the READ.RESPONSE exceeding the thresholds – which is a performance bottleneck to be analysed for further resolutions.

WRITE.RESPONSE






From the above the files and the records exceeding the write response thresholds are to be analysed for performance bottleneck.

Tmon

Tmon Analyser is a tool to analyse all the details gathered by the TecPack in a graphical manner. The Analyser lets us know about the performance of a T24 System. Tecpack is the jBase file that gets updated with the details of the transactions. This file creation and updation happens at the COB.

The details gathered by the T24 TecPack are in a raw format and analysis of the Tecpack is quite difficult. Also the information gathered by this process is too little to judge the performance of the underlying T24 system. So in-order to overcome these problems, the Tmon Analyser was developed. The Tmon Analyser records and bundles most of the information from the client's current system into a single file. On top of it, the T24 TecAnalyser makes us get a clear picture on the performance of COB over a period of time. This tool serves as a starting point for product Analysts to identify and pinpoint problem areas during a COB Run.

- TMONGEN is a server side utility that gathers required T24 information and dumps it into a .tom file.

- TMON Analuser

  The Tmon Analyser Master transforms the .tmon into a Full-Fledge spreadsheet macro for further analysis. The .tmon file serves as an input for this utility. This tool itself serves as the target report for navigation of report and analysis.

**Formatted:** Normal

**Formatted:** List Paragraph, Bulleted + Level: 1 + Aligned at: 0.63 cm + Indent at: 1.27 cm

**Formatted:** Normal

**Formatted:** List Paragraph, Bulleted + Level: 1 + Aligned at: 0.63 cm + Indent at: 1.27 cm

**Formatted:** No bullets or numbering

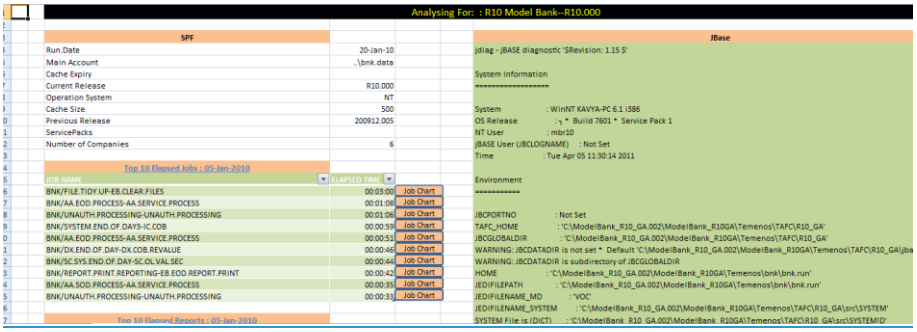
The link to download the Tmon Analyser is - http://knowledgebase.temenosgroup.com/Pages/tools.aspx

The Tmon Report contains multiple sheets which contain the report output. These are broadly classified into:

- Parameter Sheet
- Summary
- Data Sheets
- Chart Sheets
- Temp

Formatted: Normal

Summary




The summary sheet provides information as below:

- COB Run date, T24 Release, Operating System, Cache Size, number of companies etc.
- jDiag Output
- Top 10 COB Jobs (This has a link to the JobChart sheet where the Job Times detail, throughput/time elapsed, throughput/records processed, throughput/total agents are displayed in the graphical representation)
- Top 10 Report Jobs (This has a link to the Report chart where the report times detail, time elapsed for this report job for the past 5 days is displayed as a graphical representation)
- Total Lock Collisions
- COB timing stage wise and total COB time.
- Other info like the SPF, EB.EOD.ERROR, EB.SYSTEM.SUMMARY are also displayed in separate sheets.

Formatted: List Paragraph, Bulleted + Level: 1 + Aligned at: 1.27 cm + Tab after: 1.9 cm + Indent at: 1.9 cm

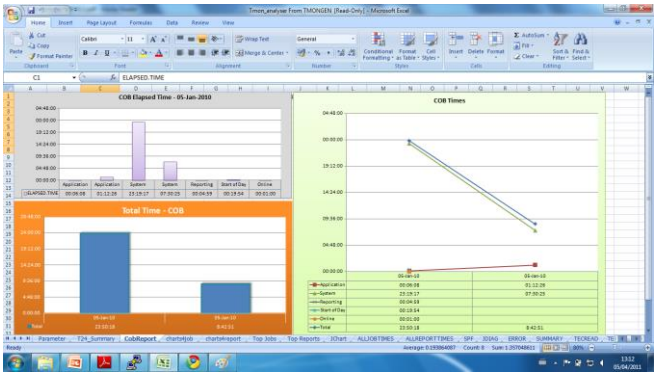
Formatted: Normal

COB Report

This sheet provides the details of the COB timings for the different stages in a graphical representation

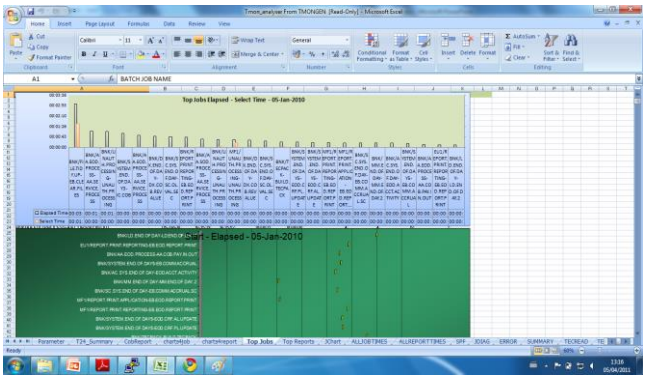
Formatted: No bullets or numbering

Formatted: Normal

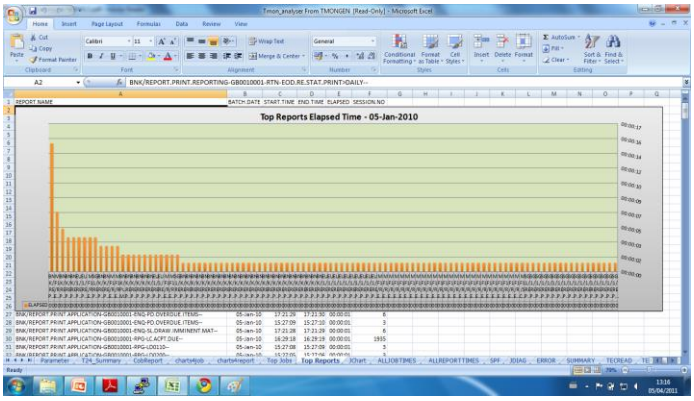



Top Jobs sheet provides graphical representation for the top job timings and the respective select timings.

Formatted: No bullets or numbering



Top Reports sheet provides graphical representation of the report timings.




Formatted: Normal

## Protocol

Frequently checking F.PROTOCOL file can give some information about T24 system health. PROTOCOL will give information like LICENCE errors, how often T24 cache is being rebuilt, the reason for SECURITY.VIOLATION messages in browser will be updated in PROTOCOL, issues regarding browser tokens etc

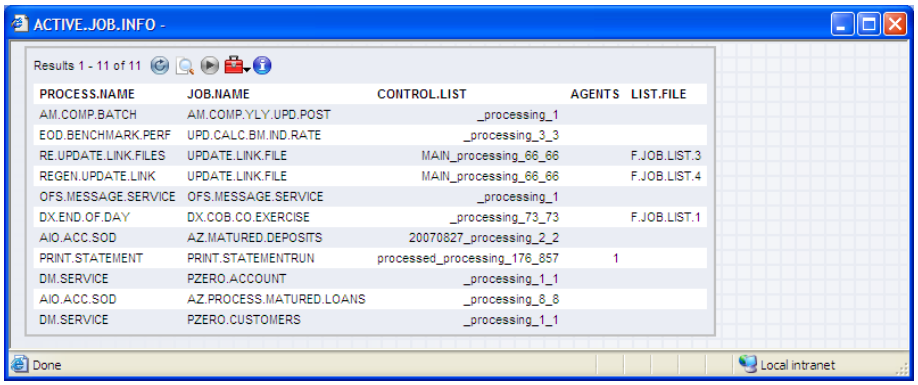
## Close of Business

COB.MONITOR can be used to monitor the close of business, COB.MONITOR gives details about how many jobs completed, how many jobs pending and percentage of each etc

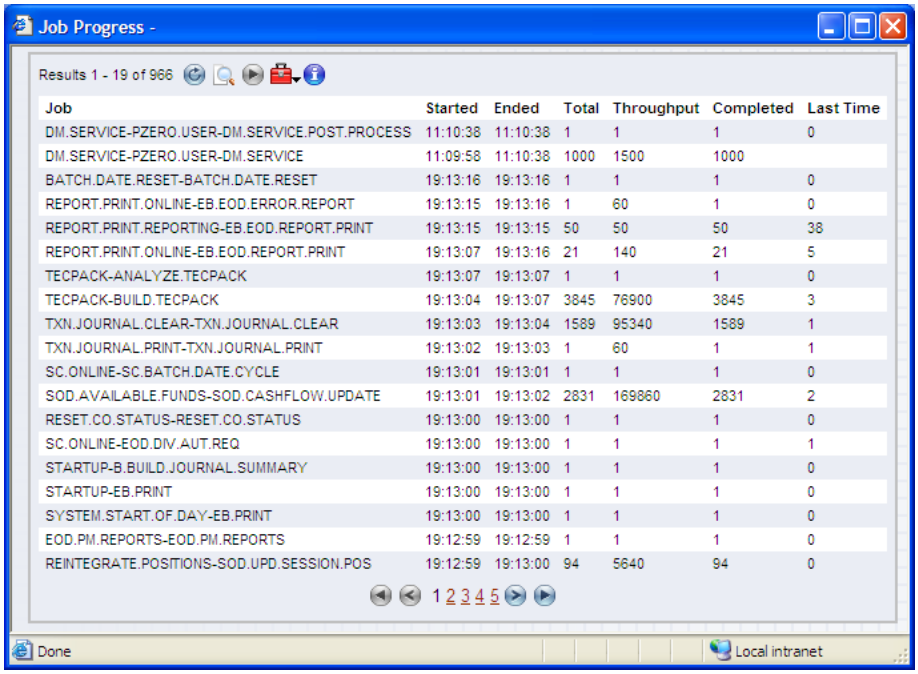
To monitor what jobs are running currently T24 core provides some enquiries called ACTIVE.JOB.INFO and JOB.PROGRESS

ACTIVE.JOB.INFO gives details about the jobs which are getting processed, this enquiry also gives details about what JOB.LIST files are getting used currently

Please note that COB.MONITOR is a tool in CUI on classic view. To monitor the COB in web browser in rich graphical view use T24 Enterprise monitor.

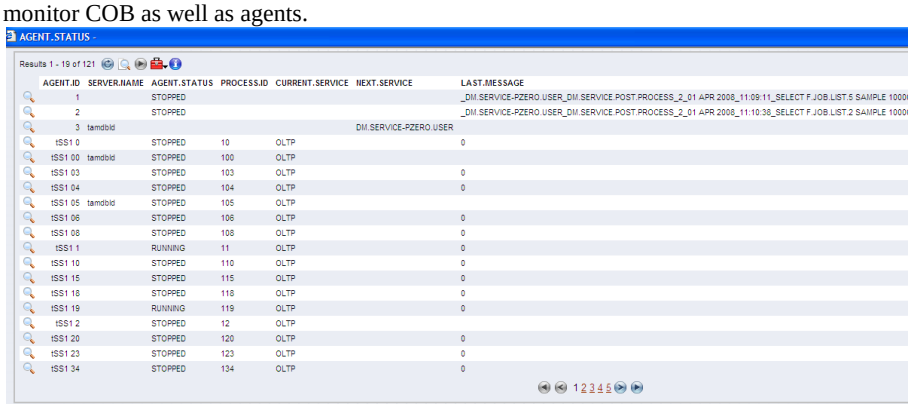



JOB.PROGRESS enquiry gives details about each and every job in the COB, it shows when the job started, finished, how many records processed, what's the throughput and elapsed time



## TSM & TSA

AGENT.STATUS enquiry gives the details about the TSM and tSA running in T24, F.TSA.STATUS file holds all the current information about TSM and tSA which are running currently. Frequently checking F.TSA.STATUS or AGENT.STATUS enquiry will help to monitor COB as well as agents.



AGENT.STATUS -

Results 1 - 19 of 121

| AGENT.ID | SERVER.NAME | AGENT.STATUS | PROCESS.ID | CURRENT.SERVICE | NEXT.SERVICE | LAST.MESSAGE |
|---|---|---|---|---|---|---|
| 1 | | STOPPED | | | | _DM.SERVICE-PZERO.USER_DM.SERVICE.POST.PROCESS_2_01 APR 2008_11:09:11_SELECT F.JOB.LIST 5 SAMPLE 1000 |
| 2 | | STOPPED | | | | _DM.SERVICE-PZERO.USER_DM.SERVICE.POST.PROCESS_2_01 APR 2008_11:10:38_SELECT F.JOB.LIST 2 SAMPLE 1000 |
| 3 | tamdbld | | | | DM.SERVICE-PZERO.USER | |
| tSS1 0 | | STOPPED | 10 | OLTP | | 0 |
| tSS1 00 | tamdbld | STOPPED | 100 | OLTP | | |
| tSS1 03 | | STOPPED | 103 | OLTP | | 0 |
| tSS1 04 | | STOPPED | 104 | OLTP | | 0 |
| tSS1 05 | tamdbld | STOPPED | 105 | OLTP | | |
| tSS1 06 | | STOPPED | 106 | OLTP | | 0 |
| tSS1 08 | | STOPPED | 108 | OLTP | | 0 |
| tSS1 1 | | RUNNING | 11 | OLTP | | 0 |
| tSS1 10 | | STOPPED | 110 | OLTP | | 0 |
| tSS1 15 | | STOPPED | 115 | OLTP | | 0 |
| tSS1 18 | | STOPPED | 118 | OLTP | | 0 |
| tSS1 19 | | RUNNING | 119 | OLTP | | 0 |
| tSS1 2 | | STOPPED | 12 | OLTP | | |
| tSS1 20 | | STOPPED | 120 | OLTP | | 0 |
| tSS1 23 | | STOPPED | 123 | OLTP | | 0 |
| tSS1 34 | | STOPPED | 134 | OLTP | | 0 |

1 2 3 4 5

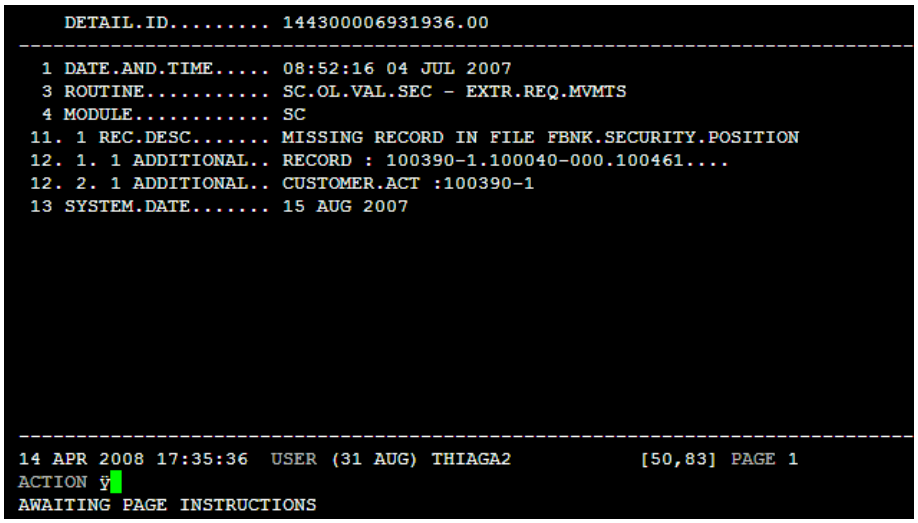
*Sample screen shot of AGENT.STATUS*

## EB.EOD.ERROR and DETAIL

All the errors during COB will be captured in EB.EOD.ERROR and EB.EOD.ERROR.DETAIL files. These two files needs to be monitored on regular basis to find out the issues occurred during COB and the way to solve them.

EB.EOD.ERROR will have information about specific problems faced during COB and .DETAIL will have more detailed information about the problem. All the issues marked as CRITICAL in EB.EOD.ERROR needs to resolved before the next COB otherwise COB will fatal out and it never allows continuing the COB.

```
    EB.EOD.ERROR.ID... GB0010001.20070830
------------------------------------------------------------------------------
  1. 1 TIME.DATE..... 19:07:32 19 MAR 2008
  2. 1. 1 DESCRIPTION. Job Dependency Error-EU.DX.CONVERT.POSN, Job Name-EU.DX.CONVERSION.TIDY, Process Name-EU.DX.CONVERT.CCY
  3. 1 APPLICATION.ID. EU.DX.CONVERT.CCY-BACK.VALUE.SUSPENSION
  4. 1 ROUTINE........ S.JOB.RUN
  5. 1. 1 RECORD.KEY.. 2 100114/101
  6. 1. 1 DETAIL.KEY.. 146890011868852.01
  8. 1 FIX.REQUIRED... YES
  1. 2 TIME.DATE..... 19:07:41 19 MAR 2008
  2. 2. 1 DESCRIPTION. Job Dependency Error-DX.COB.CO.SYSTEM, Job Name-DX.COB.CO.EXERCISE, Process Name-DX.END.OF.DAY
  3. 2 APPLICATION.ID. DX.END.OF.DAY-BACK.VALUE.SUSPENSION
  4. 2 ROUTINE........ S.JOB.RUN
  6. 2. 1 DETAIL.KEY.. 146890011868861.01
  8. 2 FIX.REQUIRED... YES
  1. 3 TIME.DATE..... 19:07:44 19 MAR 2008
  2. 3. 1 DESCRIPTION. Job Dependency Error-DX.COB.CO.SYSTEM, Job Name-DX.COB.CO.EXERCISE, Process Name-DX.END.OF.DAY
  3. 3 APPLICATION.ID. DX.END.OF.DAY-BACK.VALUE.SUSPENSION
------------------------------------------------------------------------------
14 APR 2008 17:18:53  USER (31 AUG) THIAGA2          [50,83] PAGE 1   >>17>>>
ACTION
AWAITING PAGE INSTRUCTIONS
```




### Lock Collisions

For T24 this is the single most common cause of performance degradation, and it is critical that these lock collisions are minimised.

Lock collisions can be found out by frequently checking F.TEC.OUTPUT file, by default from R8 lock collisions will be captured by TEC with the use of default TEC.PROFILE.

Use TEC.MONITOR or LIST F.TEC.OUTPUT LIKE …LOCK… command to see all the lock collisions during online or COB.

```
LIST F.TEC.OUTPUT LIKE ...LOCK... F2 F10

F.TEC.OUTPUT..     F2...........     F10............

56.1.LOCK.COLL     0001000301        F.ACCOUNT
ISION
72.1.LOCK.COLL     0004000197        F.ACCOUNT
ISION
                   0001000372        F.ACCOUNT
63.1.LOCK.COLL     0004000106        F.ACCOUNT
ISION
28.1.LOCK.COLL     FBNK.CONSOL.U     AC.1.TR.USD.500
ISION              PDATE.WORK        1.3001.US.....3
                                     100*CREDIT**200
                                     70831*
57.2.LOCK.COLL     FBNK.CONSOL.U     AC.1.TR.USD.500
ISION              PDATE.WORK        1.3001.BE.....3
                                     100*CREDIT**200
                                     70831*
                   0004000115        F.ACCOUNT
32.1.LOCK.COLL     FBNK.CONSOL.U     AC.1.TR.USD.500
ISION              PDATE.WORK        1.3001.BE.....3
                                     100*CREDIT**200
                                     70831*
24.1.LOCK.COLL     FBNK.CONSOL.U     AC.1.TR.SGD.500
ISION              PDATE.WORK        1.3001.SG.....3
                                     100*CREDIT**200
                                     70831*
27.2.LOCK.COLL     FBNK.CONSOL.U     AC.1.TR.USD.500
ISION              PDATE.WORK        1.3001.BE.....3
                                     100*CREDIT**200
                                     70831*
```

Release before R8, TEC.PROFILE needs to be enabled for LOCK.COLLISIONS for all users to get results in F.TEC.OUTPUT file. Please refer TEC user guide on how to set TEC.PROFILE

### jEDI Driver log

When running in oracle as the database for T24, jedi driver log provides much useful information. By default jedi driver log (~~ORAdriver.log~~ XMLdriver.log) will get updated when jedi driver faces an error (oracle error, xml parsing error). So checking this frequently will help to monitor T24 activities with oracle and also to find any issues and solve them.

Jedi driver log can generate more data if JEDI_XMLORACLE_TRACE is set to 1 but extreme care must be taken before setting this variable as it will affect performance a lot. Only when you want to analyse the issue or to find out why the error is coming or when you want to analyse what is exactly happening from T24 to ORACLE you can enable this flag.

If JEDI_XMLORACLE_TRACE is set then jedi driver log will have all details about jql to sql conversion, time taken for each database activity (ex: read, write, delete, open, select etc) The following are the significant oracle driver variables:

JEDI_XMLDRIVER_PIDLOG=1 - Each process generates trace file based on pid
JEDI_XMLDRIVER_DEBUG=1 - Only trace Query translation
JEDI_XMLDRIVER_DEBUG_DISPLAY=1 - Display translated Query to screen
JEDI_XMLDRIVER_EXPLAIN=1 - Log explain plan for queries
JEDI_XMLDRIVER_EXPLAIN_PLAN_DISPLAY=1 - Display explain plan to screen

Formatted: Normal

## TECPack

TECPack will be built during COB by BUILD.TEC.PACK job, TECPack contains all the useful information regarding last working day of the bank.

TECPack will contain the following information

- JOB.TIMES for last 10 COB's
- REPORT.TIMES for last 10 COB's
- COB Errors from EB.EOD.ERROR and DETAIL
- TEC events for the last day like LOCK.COLLISSIONS, READ.SIZE, READ.RESPONSE, SELECT.TIME etc
- SPF
- Jdiag

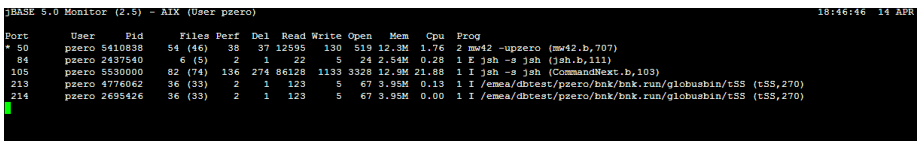
TECPack needs to be analysed on a daily basis and needs to be sent to HD when needed

## mw42

One of the best tools to monitor T24 processes is mw42, mw42 gives lot of useful information regarding T24 processes. This can be executed from jshell with some options as well.

Mw42 gives information like number of reads, writes, opens, deletes, executes, memory utilised, CPU consumed etc

Mw42 uses memory mapped files to capture the data (each jbase processes will keep the information in memory and mw42 reads that process memory and gets the information) so performance wont be impacted by using mw42 monitor.



Mw42 also gives subroutine trace which shows which subroutine and which line of code is currently been executed by jBASE, using this functionality performance issues can be easily monitored and analysed

During COB tSAs can be monitored to see where each job is spending most of the time, execute

**mw42 –u<unixuserid>|grep tSA**

by the above command specific processes can be monitored with mw42. Similarly mw42 output can be redirected to a log file as well with the use of following command

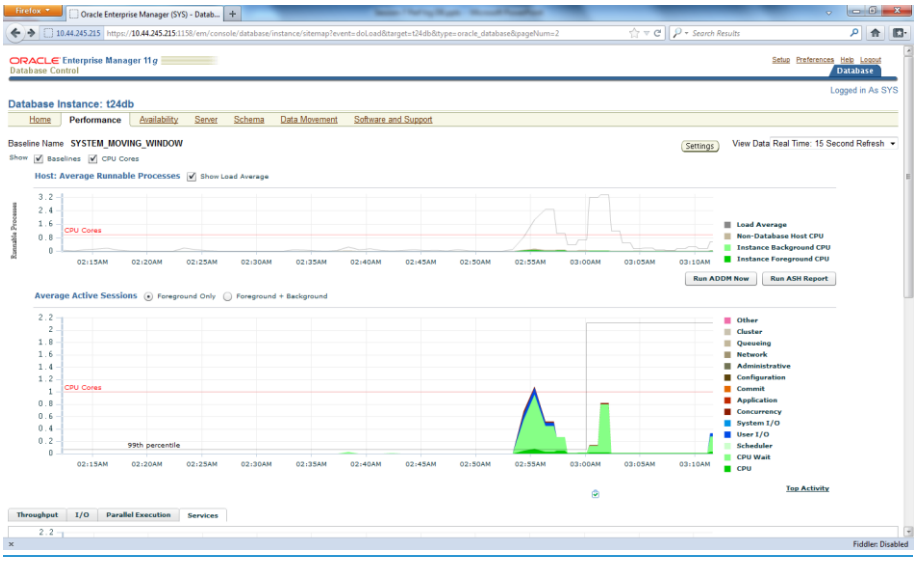
**mw42 -f -t 5 10000 | tee mw42_date_out**
*The above command will redirect mw42 output to mw42_date_out file*

Joint 2 mw42 sections

Oracle

## Configuring OEM (Enterprise Manager)

Oracle's Enterprise Manager, a Web-based Database Control, serves as the primary tool for managing an Oracle database. It is primarily used for viewing performance and status information about the database instance.




It is also possible to perform administrative tasks such as creating schema objects, managing user security, backing up and recovering the database, and importing and exporting data, although in many cases this is usually performed by DBA's using the command line interfaces.

The primary method for configuring an existing Oracle Database 10g database so it can be managed with the Database Control is to use DBCA (Database Creation Assistant). The use of DBCA requires an x windows client. If the database was built using the DBCA, OEM will automatically be configured.

If the database was built manually using scripts, the DBCA can be used afterwards, to configure OEM, or it can be done manually from the command line interface.

The AWR and ADDM reports are initiated and collected through the OEM front end

**Formatted:** Font: 12 pt, English (United States)

**Formatted:** Font: 12 pt, English (United States)

### Pre-Requirements in all cases:

- The init.ora parameter job_queue_processes=1 (or greater).
- The init.ora parameter remote_login_passwordfile=EXCLUSIVE (this is the default).
- The sysaux tablespace is larger than 100m (350m - 500m is ok).
- An oracle password file has been created file in $ORACLE_HOME/dbs, using the orpwd utility. For example:

  **orapwd file='/home/oracle/10gR2/dbs/orapwTEST1' password='test1' entries='5'**
- Ensure that the database entry exists in the oratab file.

  The oratab file is generally located in /etc or /var/opt/oracle (on Solaris) an example entry looks as follows: **TEST1:/home/oracle/10gR2:N**

### Process Using DBCA:

- Login to $ORACLE_HOME/bin and run ./dbca
- Select the configure database option
- Advance to the Management Options page and select the following options: **Configure the Database with Enterprise Manager Use, Database Control for Database Management**
- Advance until the **Finish** button is available.
- Click **Finish** to reconfigure the database so it uses Database Control.

The OEM url, for access to the database, is given on the final screen for example: http://serverx.temenos.com:1158/em

### Process: Using Command Line Interface (Manual Configuration)

(x windows is not required for this)

From the Unix command prompt type the following, ensuring ORACLE_HOME and ORACLE_SID are set.
$ **emca -config dbcontrol db -repos create**
Enter details of the database and user passwords at the prompts. The default listener port is 1521. Make sure the ORACLE_HOME and ORACLE_SID environment variables are set. Below is an example:

STARTED EMCA at 14-Feb-200811:09:28
EM Configuration Assistant, Version 10.2.0.3.0 Production
Copyright (c) 2003, 2005, Oracle. All rights reserved.
Enter the following information:
Database SID: **MBDEMO**
Listener port number: **1521**
Password for SYS user: *****
Password for DBSNMP user: *****
Password for SYSMAN user: *****
Email address for notifications (optional):
Outgoing Mail (SMTP) server for notifications (optional):
---
You have specified the following settings
Database ORACLE_HOME ..... /home/oracle/oracle/product/10.2.0/db_1
Database hostname ..... serverx.temenos.com
Listener port number ..... 1521
Database SID ..... MBDEMO
Email address for notifications .....
Outgoing Mail (SMTP) server for notifications .....
---
Do you wish to continue? [yes(Y)/no(N)]: **y**
14-Feb-200811:10:06 oracle.sysman.emcp.EMConfig perform
INFO: This operation is being logged at /home/oracle/oracle/product/10.2.0/db_1/cfgtoollogs/emca/MBDEMO/emca_2008-02-14_11-09-28-AM.log.
14-Feb-200811:10:12 oracle.sysman.emcp.EMReposConfig createRepository
INFO: Creating the EM repository (this may take a while) ...
14-Feb-200811:13:50 oracle.sysman.emcp.EMReposConfig invoke
INFO: Repository successfully created
14-Feb-200811:13:55 oracle.sysman.emcp.ParamsManager getLocalListener
14-Feb-200811:13:57 oracle.sysman.emcp.util.DBControlUtil startOMS
INFO: Starting Database Control (this may take a while) ...
14-Feb-200811:15:34 oracle.sysman.emcp.EMDBPostConfig performConfiguration
INFO: Database Control started successfully
14-Feb-200811:15:34 oracle.sysman.emcp.EMDBPostConfig performConfiguration
INFO: >>>>>>>>>>> The Database Control URL is http://serverx.temenos.com:1158/em <<
Enterprise Manager configuration completed successfully
FINISHED EMCA at 14-Feb-200811:15:34

After configuring OEM for the database, a new subdirectory appears in the ORACLE_HOME directory. This directory is named using the following format and contains Database Control configuration and state files specific to the database you just configured: hostname_sid.
Also the OEM url for access to the database is given: http://serverx.temenos.com:1158/em

The following details were added to the **portlist.ini file,** which is located in the $ORACLE_HOME/install directory:
path=/home/oracle/oracle/product/10.2.0/db_1
Ultra Search HTTP port number =5620
Enterprise Manager Agent Port =
iSQL*Plus HTTP port number =5560
**Enterprise Manager Console HTTP Port (TEST1) = 1158**
Enterprise Manager Agent Port (TEST1) = 1830

To access OEM open an Internet browser and type the url: http://hostname:port/em. If the hostname supplied does not bring up the login page, use the ip address of the server instead. If you forget the port number then it can be located from the portlist.ini file as highlighted above.

### Troubleshooting:

If you are unable to reach the OEM homepage it is probably likely that the dbconsole is not running. To check this type the following at the Unix command prompt:
**emctl status dbconsole**

To start it up, type the following and then check the status:

**emctl start dbconsole**

### What to monitor in OEM?

OEM can be really useful to monitor oracle to see what is actually happening at the database level. It will help to find out where oracle is spending most of its time

- Top wait events
- Top SQL statements which are time consuming

Other than the above it also gives lot of other information like how to tune improper sql statements, and how to improve performance etc.

### AWR report

AWR report in oracle gives plenty of useful information for monitoring performance. Using this report almost all the performance bottlenecks in oracle database can be found out. AWR report can be taken on an hourly basis from oracle database, by defaults all the information for AWR report will be captured and stored in oracle admin database by ORACLE.

Mainly the following needs to be monitored to analyse oracle performance

Under top wait events find whether you have the following

- Db sequential read

- Db scattered read
- HW contention
- Log file sync
- Row lock contention
- Sql net more data

**Top 5 Timed Events**

| Event | Waits | Time(s) | Avg Wait(ms) | % Total Call Time | Wait Class |
|---|---|---|---|---|---|
| CPU time | | 155,964 | | 54.3 | |
| log file sync | 3,549,835 | 92,418 | 26 | 32.2 | Commit |
| db file sequential read | 40,231,269 | 88,447 | 2 | 30.8 | User I/O |
| direct path read | 1,107,108 | 4,387 | 4 | 1.5 | User I/O |
| log file parallel write | 222,737 | 3,786 | 17 | 1.3 | System I/O |

## Main Report

- Report Summary
- Wait Events Statistics
- SQL Statistics
- Instance Activity Statistics
- IO Stats
- Buffer Pool Statistics
- Advisory Statistics
- Wait Statistics
- Undo Statistics
- Latch Statistics
- Segment Statistics
- Dictionary Cache Statistics
- Library Cache Statistics
- Memory Statistics
- Streams Statistics
- Resource Limit Statistics
- init.ora Parameters

Back to Top

*Sample AWR report taken from oracle enterprise manager*

### Database size

It's absolutely necessary to monitor the database size on a regular basis (weekly is recommended), this activity will help to monitor and understand oracle database growth with respective to T24. These scripts will also help to find out what T24 tables, indexes; LOB's are growing rapidly in the database.

Discuss with Oracle DBA with the results of the below scripts and plan the database tuning options for controlling the database size. Application changes will also be necessary to stabilize the database growth sometimes.

### Data File Sizes

Can run as t24 user from sqlplus:
Current Size of Data Files

```
col file_name for a40
col tablespace_name for a17
set pagesize 500

select bytes/1024/1024 as “mb”, tablespace_name,file_name from dba_data_files;
```

### Total Database Size

```
Select sum(bytes/1024/1024) as “Total mb” from dba_data_files;
```

### Current Free Space In Database

```
select sum(bytes/1024/1024) as “Total Space mb”, max(bytes/1024/1024) as “Max
Contiguous Space mb”, tablespace_name
from dba_free_space
group by tablespace_name;
```

### Object Size

Need function ‘GET_COUNT’ created  first

### Tables

Compile Get _count function:  run this in sqlplus:

```
CREATE OR REPLACE FUNCTION GET_COUNT(sTABLE IN VARCHAR) RETURN NUMBER IS
RCOUNT INT;
BEGIN
EXECUTE IMMEDIATE 'SELECT COUNT(*) FROM ' || sTABLE  INTO     RCOUNT;
 RETURN RCOUNT;
END GET_COUNT;
/
```

### Create TOPTAB Table (top 50 tables)

```
CREATE TABLE toptab AS SELECT * FROM
```

```
(SELECT segment_name, sum(bytes)/1024/1024 AS "mb"
FROM dba_segments
WHERE segment_type='TABLE'
AND owner='T24'
GROUP BY segment_name
ORDER BY 2 DESC)
WHERE ROWNUM<=50;
```

#### Run Query

```
set pagesize 500
col table for a30
SELECT SEGMENT_NAME as "table", "mb", GET_COUNT(SEGMENT_NAME) as "rows" FROM
toptab;
```

### Indexes

#### Create TOPIND Table

```
set pagesize 500
COL SEGMENT_NAME FOR A30
CREATE TABLE topind AS SELECT * FROM
(SELECT segment_name, sum(bytes)/1024/1024 AS "mb"
FROM dba_segments
WHERE segment_type='INDEX'
AND owner='T24'
GROUP BY segment_name
ORDER BY 2 DESC)
WHERE ROWNUM<=50;
```

#### Run Query

```
set pagesize 500
col index for a30
SELECT SEGMENT_NAME as "Index", "mb" FROM TOPIND;
```

### LOB

#### Create TOPLOB Table

```
set pagesize 500
COL SEGMENT_NAME FOR A30
CREATE TABLE toplob AS SELECT * FROM
(SELECT segment_name, sum(bytes)/1024/1024 AS "mb"
FROM dba_segments
WHERE segment_type='LOBSEGMENT'
AND owner='T24'
GROUP BY segment_name
```

```
ORDER BY 2 DESC)
WHERE ROWNUM<=50;
```

### Run Query

```
set pagesize 500
col lobsegment for a30
SELECT SEGMENT_NAME as "Lobsegment", "mb" FROM toplob;
```

## LOBINDEX

### Create TOPLIN Table

```
set pagesize 500
COL SEGMENT_NAME FOR A30
CREATE TABLE toplin AS SELECT * FROM
(SELECT segment_name, sum(bytes)/1024/1024 AS "mb"
FROM dba_segments
WHERE segment_type='LOBINDEX'
AND owner='T24'
GROUP BY segment_name
ORDER BY 2 DESC)
WHERE ROWNUM<=50;
```

### Run Query

```
set pagesize 500
col lobindex for a30
SELECT SEGMENT_NAME as "lobindex", "mb" FROM toplin;
```

## Redo Logs

This query will show the path where the redo logs are located. You will then need to look at the file sizes from the o/s prompt.

```
col member for a40
select * from v$logfile;
```

## Top 10 table information

For table size, row count and average row length – replace toptab table with this one:

```
CREATE TABLE toptab AS SELECT * FROM
(SELECT segment_name, sum(bytes)/1024/1024 AS "mb", avg_row_len
FROM user_segments a, user_tables b
WHERE segment_type='TABLE'
```

```
and a.segment_name=b.table_name
GROUP BY segment_name,avg_row_len
ORDER BY 2 DESC)
WHERE ROWNUM<=50;
```

### Run Query

```
set pagesize 500
col table for a30
SELECT SEGMENT_NAME as "table", "mb", GET_COUNT(SEGMENT_NAME) as "rows",
avg_row_len
FROM toptab;
```

BEFORE you create it you must run the stats to get the latest figures:

```
EXECUTE DBMS_STATS.GATHER_SCHEMA_STATS('T24',DBMS_STATS.AUTO_SAMPLE_SIZE);
```

| Default Oracle parameters |
|---|
| optimizer_index_cost_adj=1 |
| open_cursors=5000 |
| session_cached_cursors=5000 |
| cursor_sharing=EXACT (if XMLOracle Driver below version 3.5 otherwise SIMILAR unless Oracle 11.2.0.2) |
| cursor_sharing=FORCE if Oracle 11.2.0.2. |
| query_rewrite_enabled=TRUE |
| query_rewrite_integrity=TRUSTED |
| undo_management=AUTO |
| undo_retention=3600 (this can be adjusted once tuning of Selects has been completed) |
| log_checkpoint_timeout=0 |
| sga_max_size=6G (depends on volumes adjust as necessary) |
| sga_target=4G (depends on volumes adjust as necessary) |
| pga_aggregate_target=1g (adjust as necessary) |
| workarea_size_policy=AUTO |
| fast_start_mttr_target=600 ( or higher) |
| filesystemio_options=SETALL |
| For HP platforms only:<br>hpux_sched_noage=154 (for HP/UX 11.0)<br>hpux_sched_noage=178 (for HP/UX 11i) |
| For databases versions below 11gR2, set the following event to enable the fix for High Watermark Contention:<br>alter system set event = '44951 trace name context forever, level 1024' scope=spfile;<br>This requires the database to be re-started to take effect. |

The following section shows some examples of AWR and ADDM report interpretation:

Case1:
The following screen capture is extracted from a AWR report under the TOP 5 timed events:

| Event | Waits | Time(s) | Avg Wait(ms) | % Total Call Time | Wait Class |
|---|---|---|---|---|---|
| log file sync | 210,293 | 36,067 | 172 | 32.4 | Commit |
| CPU time | | 15,703 | | 14.1 | |
| db file sequential read | 1,925,155 | 11,619 | 6 | 10.4 | User I/O |
| cursor: pin S | 13,482,233 | 4,485 | 0 | 4.0 | Other |
| SQL*Net more data to client | 24,058,747 | 3,273 | 0 | 2.9 | Network |

Performance on the disks with the redo logs is the most critical for performance, from the above it is time spent on the Log file sync wait is more and not in the acceptable range.

Solution:
Average wait time for log file sync should be under 10ms. To achieve this DBA has to increase the size of the log files or increase the no of log files.

Case 2:
The following items is reported in the LOG SWITCHES section of AWR report.

| Statistic | Total | per Hour |
|---|---|---|
| log switches (derived) | 59 | 59.38 |

Log switches seem to be happening so frequent causing the overhead to the system resources.

Solution:
Resize the redo logs to switch at about every 20 minutes.

Case3:
The following is extracted from AWR report on a system, while COB is running:

**Formatted:** Font: (Default) Calibri, 11 pt, Font color: Custom Color(RGB(31,73,125))

**Formatted:** Font: (Default) Calibri, 11 pt, Font color: Custom Color(RGB(31,73,125))

**Top 5 Timed Events**

| Event | Waits | Time(s) | Avg Wait(ms) | % Total Call Time | Wait Class |
|---|---|---|---|---|---|
| enq: HW - contention | 79,906 | 33,143 | 415 | 68.7 | Configuration |
| db file sequential read | 518,791 | 5,103 | 10 | 10.6 | User I/O |
| CPU time | | 2,674 | | 5.5 | |
| RMAN backup & recovery I/O | 45,405 | 2,078 | 46 | 4.3 | System I/O |
| gc current block busy | 29,651 | 946 | 32 | 2.0 | Cluster |

There are two issues seen, one is the high water mark contention and the RMAN backup & recovery(heavy IO job scheduled at the same time as COB time)

Solution:

1. Set the following event to enable the fix for High Watermark Contention:
   alter system set event = '44951 trace name context forever, level 1024' scope=spfile;
   This requires the database to be re-started to take effect.

   Please note that the above is to be applied only for databases versions below 11gR2.
   If the contention does alleviate, then partitioning to be resorted to as the second option.

2. During the COB process additional database activity – RMAN backup & recovery seen in the system. Schedule this activity outside the COB runtime window.

**Formatted:** List Paragraph, Numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0.63 cm + Indent at: 1.27 cm

**Formatted:** Indent: First line: 1.27 cm

**Formatted:** Indent: Left: 1.27 cm

**Formatted:** Indent: First line: 1.27 cm

**Formatted:** Indent: Left: 1.27 cm

**Formatted:** Numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0.63 cm + Indent at: 1.27 cm

DB2 ~~– tba~~

*There are several methods to collect information and diagnose DB2 system performance issues. The snapshot monitor is one of the most commonly used tools to collect information in order to narrow down a problem.*

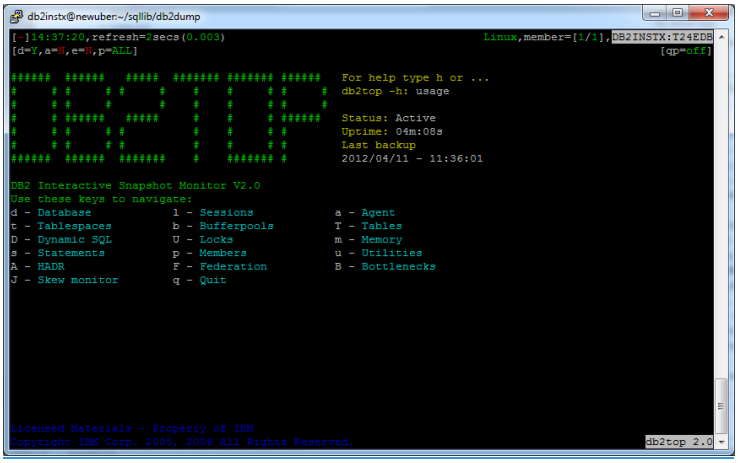
*The db2top tool comes with DB2, and can be used to calculate the delta values for those snapshot entries in real time. This tool provides a GUI under a command line mode, so that users can get a better understanding while reading each entry.*

*db2top can be run in two modes, interactive mode or batch mode.*

*Run db2top in interactive mode*

*Enter the following command from a command line to start db2top in interactive mode:*

*db2top -d T24EDB*




*Few of the key items to be monitored are described in the following section.*

**Formatted:** Normal

**Formatted:** Normal

## Database (d)

*On the database screen, db2top provides a set of performance monitoring elements for the entire database .*


***Active sessions, Sort Memory & Log space***

*Users can monitor active session (MaxActSess), sort memory (SortMemory), and log space (LogUsed). These monitoring elements can help users identify what is the current percentage of usage for those elements.*

***Lock used and Lock Escals***

*Lock usage (LockUsed) and escalation (LockEscals) can be very helpful to narrow down locking issues. If a huge number of lock escalations is observed, it is a good idea to increase the LOCKLIST and MAXLOCKS database parameters, or start looking at bad queries that may request a huge amount of locks.*

***IO***

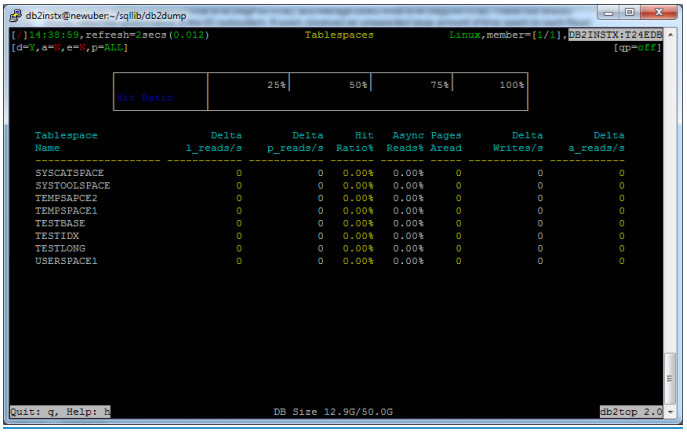
*The last four entries in the Database screen show the Average Physical Read time (AvgPRdTime), Average Direct Read Time (AvgDRdTime), Average Physical Write time (AvgPWrTime), and Average Direct Write time (AvgDWrTime). These four entries directly reflect the performance of the I/O subsystem. If users observed an unexpected large amount of time spent on each Read or Write operation, further investigation should be made into the I/O subsystem.*

**Formatted:** Font: Bold, Underline

**Formatted:** Font: Bold, Underline

**Formatted:** Font: Bold, Underline

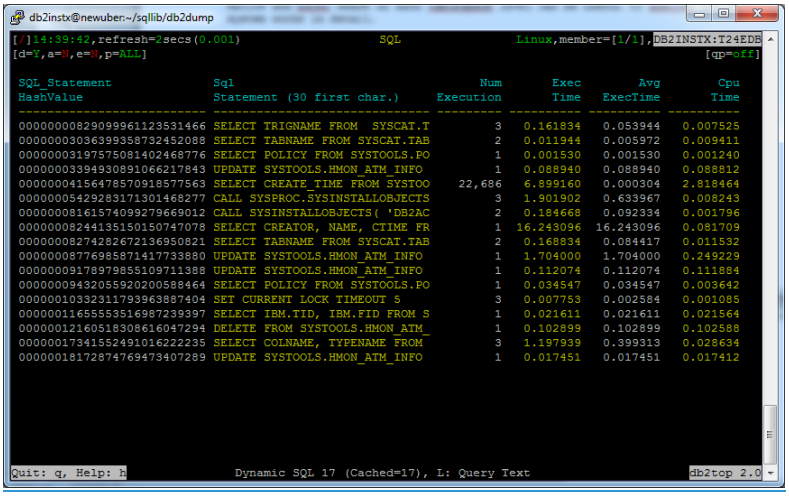
## Tablespace (t)



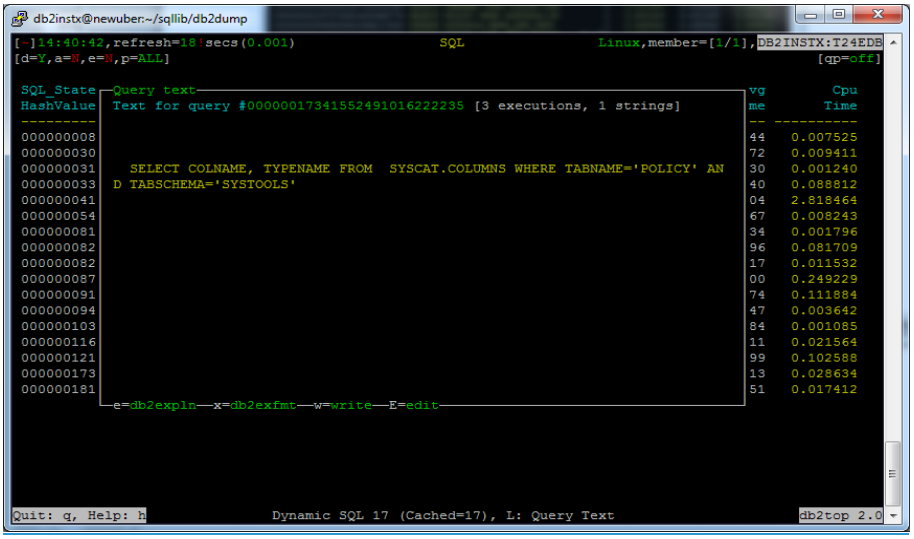

*The tablespace screen provides detailed information for each tablespace. The Hit Ratio% and Async Read% columns can be very important to many users. In an environment that contains many tablespaces, a bad query occurring in one tablespace could be obscured by averaging the hit ratio over all tablespaces. Monitoring Hit Ratio% and Async Read% on each tablespace level can be useful to analyze how a system works in detail.*

## Dynamic SQL (D)

*The Dynamic SQL screen provides detailed information for each cached SQL statement. Users can also use this screen to generate db2expln and db2exfmt output for a specific query.*

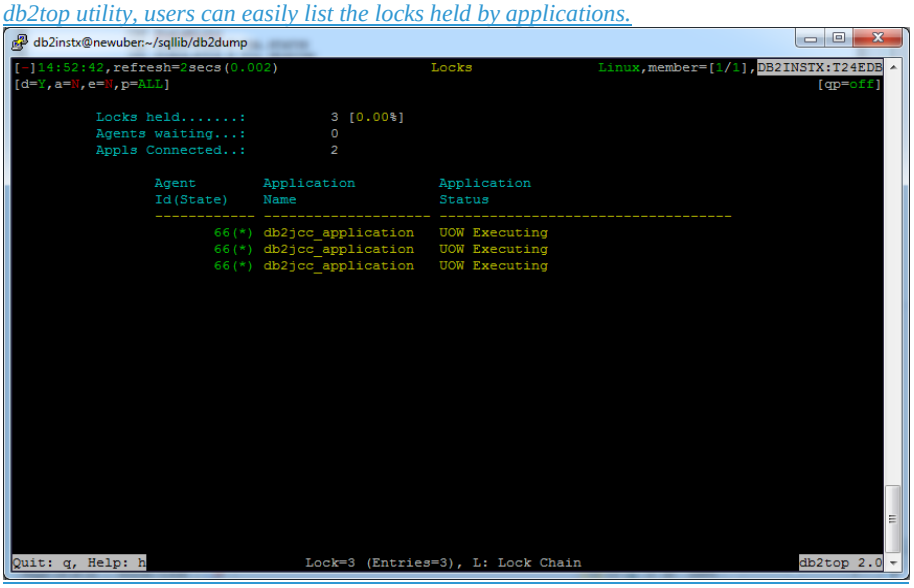


*db2top utility also provides functionality to generate a db2expln or db2exfmt report without manually running the commands. By entering a capital L on the Dynamic SQL screen, it prompts you to enter a SQL hash string. The SQL hash string is the string showing in the first column of the table, for example " 00000017341552491016222235." Users can copy the string and paste it into the prompt and click Enter, as shown in Figure:*



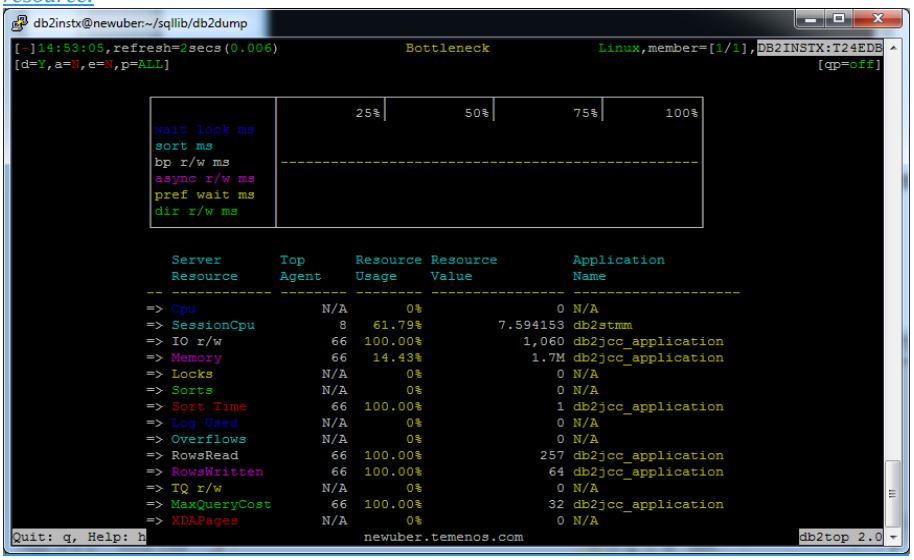
## Lock (U)

*A locking issue is one of the most commonly seen issue during application diagnosis. With db2top utility, users can easily list the locks held by applications.*




## Bottlenecks (B)

*The main body of the "Bottleneck" screen shows which agent is the bottleneck in each server resource.*



*The first column, Server Resource, in the screen "Bottlenecks" shows what kind of server resource is monitored:*

*•Cpu: Which agent consumes the most CPU time.*
*•SessionCpu: Which application session consumes the most CPU time.*
*•IO r/w: Which agent consumes the most I/O read and write.*
*•Memory: Which agent consumes the most memory.*
*•Lock: Which agent is holding the most locks.*
*•Sorts: Which agent has executed the biggest number of sorting.*
*•Sort Times: Which agent consumes the longest sorting time.*
*•Log Used: Which agent consumes the most log space in the most recent unit of work.*
*•Overflows: Which agent has the most number of sort overflows.*
*•RowsRead: Which agent has read the most number of rows of records.*
*•RowsWritten: Which agent has written the most number of rows of records.*
*•TQ r/w: Which agent has sent and received most number of rows on table queues.*
*•MaxQueryCost: Which agent has the max SQL execution time estimated by the compiler.*
*•XDAPages: Which agent has the most number of pages for XDA data (available in V9.1GA and after releases.*

SQL server - monitoring
Dammika or TED or KK

WEB Server

## Apache

### Log files

Apache log files can be found under root_directory/logs, where root_directory is tomcat installation directory.

There are many log files available under the logs directory for monitoring apache
Ex: Catalina.out
Stderr_<date>.log
<servername>.log

These log files provide very useful information about apache tomcat and its performance.

Apache tomcat logging is controlled by logging.properties file under root_directory/conf/ directory. By default INFO level logging is enabled in apache tomcat, to get more useful data and also to check system errors etc DEBUG level should be used

Web Application Server

Formatted: Font: Times New Roman

## Web Sphere

### Log files

Websphere log files can be found under root_directory/logs, where root_directory is websphere installation directory/home directory

Log files give lot of information about IBM websphere and also the application which is running on websphere. Useful system information like thread size, heap memory utilisation, http errors, application errors etc can be found in the log files which will help to monitor and tune websphere applications and parameters.

Jboss

WEB logic

Performacne tunning in general webserver
IBM WAS performance pages,
Garbage collection,
JVM,
Components monitoring,
Min & Max session – t24-ds.xml.
Queue – DLQ, monitoring the Queue stack during the peak volume and increase the threads, etc,.

### Viewing current performance activity

You can view the current performance activity of a server using the Tivoli Performance Viewer (TPV) in the administrative console.

**Before you begin**

Once monitoring is enabled, TPV monitors the performance activity of all servers on a node, which includes the associated application servers and the node agent for the node being monitored.

TPV enables administrators and programmers to monitor the current health of WebSphere Application Server. Because the collection and viewing of data occurs in the application server, performance may be affected. To minimize performance impacts, monitor only those servers whose resources need to be optimized.

**Procedure**

1. Click **Monitoring and Tuning > Performance Viewer > Current Activity** in the console navigation tree. The TPV current activity collection is displayed.

2. Start monitoring the current activity of a server in either of two ways:

    - Under **Server**, click the name of the server whose activity you want to monitor. Clicking on the name starts the monitoring for the server and displays the activity page for the server.
    - Select the check box for the server whose activity you want to monitor, and click **Start Monitoring**. To start monitoring multiple servers at the same time, select the servers and click **Start Monitoring**.

    A TPV console panel is displayed, providing a navigation tree on the left and a view of real-time data on the current performance activity of a server on the right.

3. From the navigation tree, select the server activity data that you want to view.

| Option | Description |
|---|---|
| **Advisor** | Use the Performance Advisor to examine various data while your application is running. The Performance Advisor provides advice to help tune systems for optimal performance and gives recommendations on inefficient settings by using collected PMI data. |

| Option | Description |
|---|---|
| **Settings** | Configure user and logging settings for TPV. These settings can affect the performance of your application server. |
| **Summary Reports** | View summary reports on servlets, enterprise beans (EJBs), EJB methods, connections pools and thread pools in WebSphere Application Server. |
| **Performance Modules** | View performance modules that provide graphics and charts of various performance data on system resources such as CPU utilization, on WebSphere pools and queues such as database connection pools, and on customer application data such as servlet response time. In addition to providing a viewer for performance data, TPV enables you to view data for other products or customer applications that have implemented custom PMI. |

**What to do next**
When you finish monitoring a server, select the server and click **Stop Monitoring**. TPV automatically stops monitoring a server if you the browser window are closed or if the user logs out.

More information on how to monitor IBM websphere is available in the below link

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/tprf_tpvmonitor.html

### Using the Runtime Performance Advisor to monitor performance

The advisors analyze the Performance Monitoring Infrastructure (PMI) data of WebSphere Application Server using general performance principles, best practices, and WebSphere Application Server-specific rules for tuning.

The Runtime Performance Advisor provides advice to help tune systems for optimal performance and is configured using the WebSphere Application Server administrative console or the wsadmin tool . The Runtime Performance Advisor uses Performance Monitoring Infrastructure (PMI) data to provide recommendations for performance tuning.

Running in the Java virtual machine (JVM) of the application server, this advisor periodically checks for inefficient settings, and issues recommendations as standard product warning messages. View these recommendations by clicking **Troubleshooting > Runtime Messages > Runtime Warning** in the administrative console. Enabling the Runtime Performance Advisor has minimal system performance impact.

**Procedure**

1. Ensure that PMI is enabled, which is default. If PMI is disabled, consult the Enabling PMI using the administrative console topic. To obtain advice, you must first enable PMI through the administrative console and restart the server. The Runtime Performance Advisor enables the appropriate monitoring counter levels for all enabled advice when PMI is enabled. If specific counters exist that are not wanted, or when disabling the Runtime Performance Advisor, you might want to disable PMI or the counters that the Runtime Performance Advisor enabled.

2. If running Network Deployment, you must enable PMI on both the server and the node agent, and restart the server and the node agent.

3. Click **Servers** > **Application servers** in the administrative console navigation tree.

4. Click *server_name* > **Runtime Performance Advisor Configuration**.

5. Under the **Configuration** tab, specify the number of processors on the server. This setting is critical to ensure accurate advice for the specific configuration of the system.

6. Select the **Calculation Interval**. PMI data is taken over time and averaged to provide advice. The calculation interval specifies the length of time over which data is taken for this advice. Therefore, details within the advice messages display as averages over this interval.

7. Select the **Maximum Warning Sequence**. The maximum warning sequence refers to the number of consecutive warnings that are issued before the threshold is updated. For example, if the maximum warning sequence is set to 3, then the advisor sends only three warnings, to indicate that the prepared statement cache is overflowing. After three warnings, a new alert is issued only if the rate of discards exceeds the new threshold setting.

8. Specify **Minimum CPU for Working System**. The minimum central processing unit (CPU) for a working system refers to the CPU level that indicates an application server is under production load. Or, if you want to tune your application server for peak production loads that range from 50-90% CPU utilization, set this value to 50. If the CPU is below this value, some diagnostic and performance advice are still issued. For example, regardless of the CPU level if you are discarding prepared statements at a high rate, you are notified.

9. Specify **CPU Saturated**. The CPU saturated level indicates at what level the CPU is considered fully utilized. The level determines when concurrency rules no longer increase thread pools or other resources, even if they are fully utilized.

10. Click **Apply**.

11. Click **Save**.

12. Click the **Runtime** tab.

13. Click **Restart**. Select **Restart** on the Runtime tab to reinitialize the Runtime Performance Advisor using the last configuration information that is saved to disk.

    This action also resets the state of the Runtime Performance Advisor. For example, the current warning count is reset to zero (0) for each message.

14. Simulate a production level load. If you use the Runtime Performance Advisor in a test environment, do any other tuning for performance, or simulate a realistic production load for your application. The application must run this load without errors. This simulation includes numbers of concurrent users typical of peak periods, and drives system resources, for example, CPU and memory, to the levels that are expected in production. The Runtime Performance Advisor provides advice when CPU utilization exceeds a sufficiently high level only. For a list of IBM business partners that provide tools to drive this type of load, see the topic, Performance: Resources for learning in the subsection of Monitoring performance with third-party tools.

15. Select the check box to enable the Runtime Performance Advisor.

    **Tip:** To achieve the best results for performance tuning, enable the Runtime Performance Advisor when a stable production-level load is applied.

16. Click **OK**.

17. Select **Runtime Warnings** in the administrative console under the Runtime Messages in the Status panel or look in the SystemOut.log file, which is located in the following directory:

    *profile_root*/logs/*server_name*
    Some messages are not issued immediately.

18. Update the product configuration for improved performance, based on advice. Although the performance advisors attempt to distinguish between loaded and idle conditions, misleading advice might be issued if the advisor is enabled while the system is ramping up or down. This result is especially likely when running short tests. Although the advice helps in most configurations, there might be situations

where the advice hinders performance. Because of these conditions, advice is not guaranteed. Therefore, test the environment with the updated configuration to ensure that it functions and performs better than the previous configuration.

Over time, the advisor might issue differing advice. The differing advice is due to load fluctuations and the runtime state. When differing advice is received, you need to look at all advice and the time period over which it is issued. Advice is taken during the time that most closely represents the peak production load.

Performance tuning is an iterative process. After applying advice, simulate a production load, update the configuration that is based on the advice, and retest for improved performance. This procedure is continued until optimal performance is achieved.

Note:- Refer IBM website for more information regarding performance monitoring tools

Oracle Application Server - tba

## CHECK THE DOC FOR THIS ITEM IN CONSULTANTS CORNER

# TUNING

Overview
There are no magic formulae to determine the best possible performance tuning settings. However, this section is intended as a guide
Browser

```
<parameter>
        <parameterName>Use Transformer</parameterName>
        <parameterValue>NO</parameterValue>
        <!-- Options: YES / NO -->
</parameter>
<parameter>
        <parameterName>TC_TIME_OUT</parameterName>
        <parameterValue>120</parameterValue>
</parameter>
```

TC_TIME_OUT is the time out settings that the browser servlet will wait for a response from the connector.

To allow parallel request to get processed per user session browserparameters.xml was changed to have INSTANCE as Server Connection Method

```
<parameter>
    <parameterName>Server Connection
Method</parameterName>
    <parameterValue>INSTANCE</parameterValue>
    <!-- Options: T24CONNECTOR / INSTANCE -->
</parameter>
<parameter>
    <parameterName>Instance</parameterName>
    <parameterValue>production</parameterValue>
</parameter>
```

By default it will have TCClient which will not allow parallel request processing per user session and also it wont allow TCclient level fail over

Connector - TCServer

## Adapter configuration

Configuring the correct number sessions in TC Server adapter makes huge difference in TCServer performance

```
    <ADAPTER id="T24">
<!--
      <REQUEST_FORMATTER>Thai2UTF, OFSML, UTF2Thai</REQUEST_FORMATTER>
      <RESPONSE_FORMATTER>Thai2UTF, OFSML, UTF2Thai</RESPONSE_FORMATTER>
-->
      <MAX_SESSION> 5 </MAX_SESSION>
      <MIN_SESSION> 1 </MIN_SESSION>
      <TIMEOUT>30</TIMEOUT>
      <LOGIN_CONTEXT></LOGIN_CONTEXT>
      <STARTIN></STARTIN>
      <JBASEPATH>c:\jbase5</JBASEPATH>
      <PROGRAM>tSS</PROGRAM>
      <PARAMETER>GCS</PARAMETER>
    </ADAPTER>
```

- Always use less TIMEOUT (recommended value is 30seconds)
- Try to avoid using formatters, which might slow down the performance of TCServer
- Always try to keep MIN_SESSION and MAX_SESSION as same for highly used adapter (T24 browser adapter), since adjusting sessions like scaling up and scaling down takes time in TCServer and also takes some resources in T24 application server. If you want to set up MIN_SESSION and MAX_SESSION as different then atleast increase the SCALEDOWNSPEED

  <SCALEDOWNSPEED>120</SCALEDOWNSPEED>

  Scaledownspeed is the one which determines whether idle sessions needs to be killed after a specified time or not.

## Configuring Listeners

- If MQ listener is used to process the messages then NBTHREADS tag should be set to 3times more then the number of sessions set up in MAX_SESSIONS in the adapter to get optimum performance with TCServer and T24. This setup will make sure that all the tSS sessions are utilised to the maximum and get better throughput
- Avoid using request/response formatters in listeners, which might slow down the performance of TCServer

- If you are pushing messages from T24 to MQ using BasicMQBridge component then setup mqb.properties file properly

  Keep an high value for mqb.pool.size to avoid repeated connection opening to MQ

```
mqb.pool.size=2
#
# after how many seconds of inactivity the connection on the
# Queue Manager will be closed.
#
mqb.pool.expiration=60
```

### Configuring log files:

- Avoid using DEBUG or ALL option for any adapters/listeners and its recommended to keep the log level as ERROR
- Point the log files (log.directory) to the faster disk sub system if file appender is used

  #file appender

  #

  log4j.appender.file=org.apache.log4j.RollingFileAppender

  log4j.appender.file.layout=org.apache.log4j.PatternLayout

  log4j.appender.file.layout.ConversionPattern=%d [%t] %-5p %c %x - %m%n

  log4j.appender.file.File=<log.directory>/global.log

  log4j.appender.file.MaxFileSize=1024KB

  log4j.appender.file.MaxBackupIndex=10
- Keep the MaxFileSize as 1MB and MaxBackupIndex as less as possible

Connection pooling, Min and Max sessions in JBoss with TOCF(EE)

T24

## Analysing SEAT

### Convert SEAT Info

In order to generate the Captured Seat results output to Excel format, the following needs to be done

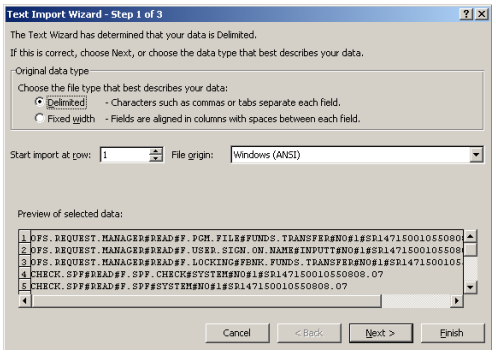
#### Converting the I/O information into Excel

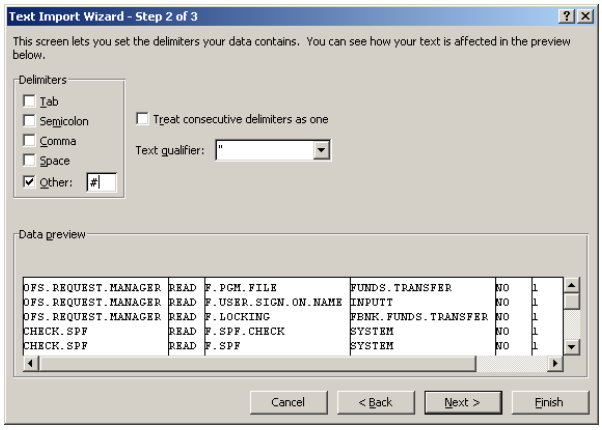
- ✓ From jShell enter the following to display the # formatted output of the Seat record
  - SEAT.IO <seat record Id>

  This displays the Seat results record in # separated Values format. The display would look something similar to the below screenshot.

```
jsh pzero ~ -->SEAT.IO SR147150010550808.07
From  just=
OFS.REQUEST.MANAGER#READ#F.PGM.FILE#FUNDS.TRANSFER#NO#1#SR147150010550808.07
OFS.REQUEST.MANAGER#READ#F.USER.SIGN.ON.NAME#INPUTT#NO#1#SR147150010550808.07
OFS.REQUEST.MANAGER#READ#F.LOCKING#FBNK.FUNDS.TRANSFER#NO#1#SR147150010550808.07
CHECK.SPF#READ#F.SPF.CHECK#SYSTEM#NO#1#SR147150010550808.07
CHECK.SPF#READ#F.SPF#SYSTEM#NO#1#SR147150010550808.07
LOG.WRITE#RELEASE#F.RECORD.LOCK#FBNK.FUNDS.TRANSFER$NAU.FT0724300233##1#SR147150010550808.07
LOG.WRITE#DELETE#F.RECORD.LOCK#FBNK.FUNDS.TRANSFER$NAU.FT0724300233##1#SR147150010550808.07
LOG.WRITE#RELEASE#F.PROTOCOL#20080414001055080S.05##1#SR147150010550808.07
LOG.WRITE#WRITE#F.PROTOCOL#20080414001055080S.05#NO#1#SR147150010550808.07
LOG.WRITE#RELEASE#F.PROTOCOL#INPUTTER##2#SR147150010550808.07
```

- ✓ Copy the entire contents from the screen and paste it into MS-Excel Spreadsheet.
- ✓ In MS-Excel, Convert the text with # delimiter to columns.






✓ Enter the headings for the Columns accordingly. The final sheet after formatting and applying Filter would be as follows:

| ROUTINE | I/O | FILE | ID | Lock | I/O Count |
|---|---|---|---|---|---|
| OFS.REQUEST.MANAGER | READ | F.PGM.FILE | FUNDS.TRANSFER | NO | 1 |
| OFS.REQUEST.MANAGER | READ | F.USER.SIGN.ON.NAME | INPUTT | NO | 1 |
| OFS.REQUEST.MANAGER | READ | F.LOCKING | FBNK.FUNDS.TRANSFER | NO | 1 |
| CHECK.SPF | READ | F.SPF.CHECK | SYSTEM | NO | 1 |
| CHECK.SPF | READ | F.SPF | SYSTEM | NO | 1 |
| LOG.WRITE | RELEASE | F.RECORD.LOCK | FBNK.FUNDS.TRANSFER$NAU.FT0724300233 | | 1 |
| LOG.WRITE | DELETE | F.RECORD.LOCK | FBNK.FUNDS.TRANSFER$NAU.FT0724300233 | | 1 |
| LOG.WRITE | RELEASE | F.PROTOCOL | 2.00804E+17 | | 1 |
| LOG.WRITE | WRITE | F.PROTOCOL | 2.00804E+17 | NO | 1 |
| LOG.WRITE | RELEASE | F.PROTOCOL | INPUTTER | | 2 |

✓ Save this into a Local File as SeatResults_Date.xls. This sheet will be used for the analysis of the Funds Transfer transaction.

### Converting the Summary information into Excel

✓ From jShell enter the following to display the # formatted output of the Seat record

- SEAT.IO <seat record Id> F.SEAT.RESULTS –DISPLAY=SUMMARY

This displays the Seat results record in # separated Values format and would contain only the summary of the entire transaction.

The display would look something similar to the below screenshot.

```
jsh pzero ~ -->SEAT.IO SR147150010550808.07 F.SEAT.RESULTS -DISPLAY=SUMMARY
From F.SEAT.RESULTS just=
EB.TRANS#1362#6#0.13#0
F.READ#8190#104#0.07#0
GET.STANDARD.SELECTION.DETS#43236#111#0.05#0
SEQX#24969#1189#0.04#0
TEC.RECORD.ACTIVITY#21357#317#0.02#0
OFS.REQUEST.MANAGER#15644#1#0.01#3
OPF#83118#716#0.1#0
System.clearVariables#20#1#0.01#0
CHECK.SPF#5120#1#0.01#2
PERFORM.ENCRYPT#22063#56#0.01#0
```

- ✓ Copy the contents and paste into a Separate Sheet in Ms-Excel.
- ✓ Convert the "#" to Columns using the procedure details above.
- ✓ Enter the headings for the Columns accordingly. The sheet would look similar to the one in the screenshot:

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | ROUTINE | PathLength | Iterations | Elapsed | Total I/O |
| 2 | EB.TRANS | 1362 | 6 | 0.13 | 0 |
| 3 | F.READ | 8190 | 104 | 0.07 | 0 |
| 4 | GET.STANDARD.SELECTION.DETS | 43236 | 111 | 0.05 | 0 |
| 5 | SEQX | 24969 | 1189 | 0.04 | 0 |
| 6 | TEC.RECORD.ACTIVITY | 21357 | 317 | 0.02 | 0 |
| 7 | OFS.REQUEST.MANAGER | 15644 | 1 | 0.01 | 3 |
| 8 | OPF | 83118 | 716 | 0.1 | 0 |
| 9 | System.clearVariables | 20 | 1 | 0.01 | 0 |
| 10 | CHECK.SPF | 5120 | 1 | 0.01 | 2 |
| 11 | PERFORM.ENCRYPT | 22063 | 56 | 0.01 | 0 |

## Analysis of SEAT results

### Analyze SEAT.RESULTS for a particular transaction

In order to analyse seat results for a particular transaction, the SEAT of the required transaction needs to be captured and converted to Ms-Excel Sheet. This is detailed in the previous section.
The sample Excel Sheet for the Funds transfer transaction is attached for reference.

SEAT Funds Transfer

The above SEAT information contains the following for the entire transaction

- For every Routine Traversed
- The Type of Input/Output
- File accessed for the I/O
- The Record ID for the corresponding File
- Whether the Record Was Locked or Not
- Number of times the File was accessed.

The below sheet provides summarised information on the following

- For every Routine
- elapsed time
- Path length
- Iterations

SEAT Summary - Funds Transfer

### Tips to Improve performance

1. Transactions either COB or online which exceed 100 I/Os are likely to prove expensive – especially in an RDBMS environment.
2. Look at each I/O – any STATIC data that is being read and the SAME record will be repeatedly read for every transaction of this type should be cached using CACHE.READ.
3. Eliminate executes – especially SELECTS (you can use CACHE.READ for selects too)
4. Try to reduce the Writes as much as possible.
5. While analyzing the I/O, look out for records which get locked and which could be locked by another user doing the same type of transaction. Lock contention destroys scalability.

### Scenarios

Few Likely scenarios where a code change might be required on analysing the Seat Results

Problem: Multiple physical Reads found for the same file and record but without a write.
Sol: Change the READ statements to call F.READ instead. For parameter or Static Files, try using CACHE.READ instead of F.READ.

Problem: Unnecessary READUs. i.e. READU done but not a WRITE.
Sol: Replace the READU with READ as there is no reason to take the Lock.

Problem: Multiple direct WRITEs to the same file and record.
Sol: Try reducing the WRITEs at multiple points by caching them and then doing one single WRITE alone.

Problem: Remove un-necessary I/O.
Sol: Finally, after all the analysis, check every I/O to satisfy the given scenario. i.e. whether the updated to the file is required or not.

Similarly the Seat Results can be captured and analysed for all applications with above given procedure and this has proved to be the best tuning option available for T24 online or COB processing.

## Resizing Files

T24 table size needs to be monitored on a regular basis (weekly basis is recommended), for this purpose T24 core provides 3 distinctive enquiries to do the job. These enquiries check all the T24 tables and giving sizing recommendation for each one of those if necessary.

EB.BAD.SIZE.FILES
EB.BAD.SIZE.FILES.IMPORTANT
EB.FILE.SIZE.HISTORY

These enquiries need to run and output needs to be analyzed to get optimum performance with T24 when jBASE is used as the database.

## Archiving – tba

## Parameter Settings for High volume clients

Check the latest changes done for R12/R13 and amednd accordingly

### ACCOUNT.PARAMETER

UPD.STMT.VAL.ENTRY = NO
This will suppress the update to STMT.VAL.ENTRY table

NEW.ACCT.PROCESS = Y
This will set the asset type of an account to sign of the first movement rather than leaving the COB process to determine the asset type

MAX.TAX.SUFFIX = 999
During tax accounts processing this will make sure no lock collision will happen during COB. Maximum 999 tax accounts will be used to avoid lock collisions

UPDATE.ENTRIES = RECENT or NONE
If value RECENT is set up then maximum of 100 entries (STMT, CATEG) will be stored in EB.CONTRACT.BALANCES file which will help to display information in TXN.ENTRY enquiry. But in case of high volume sites this is advised to setup as NONE, this will make sure no entry ids will be updated in EB.CONTRACT.BALANCES

INT.MVMT.UPDATE = NO
This will suppress the update to all INTEREST.MVMT file updates during CAPITALISATION process

CASHFLOW.DAYS = null and CREDIT.CHECK = WORKING
This will block the update of the available funds ladder for non Nostro accounts.
This will help to improve the performance of SOD.CASHFLOW.UPDATE job.

ENT.TODAY.UPDATE = NO
This is currently an NOINPUT field and an undocumented feature but in upcoming releases of T24 this needs to be set to NO from model bank itself. This setting will block the update of ACCT.ENT.TODAY, CATEG.ENT.TODAY files which will help to improve the performance of EOD.ACCT.ACTIVITY, SC.EOD.FUND.FLOW and BV.GET.TRANS and other jobs in COB.

### AC.CONSOLIDATE.COND

Make sure DEFAULT, PLDEFAULT AND REDEFAULT records are authorised which will help to consolidate all entries for internal and nostro accounts

NO.ENTRIES.START in all records is set to 1 to ensure that entries are consolidated.

For interest and charges, the detailed consolidated entries that get produced are suppressed.

IC, AC, SC & FT in the field NO.DETAIL.APP in DEFAULT, REDEFAULT & PLDEFAULT records

FILTER.SYSID field should be set to ALL
FILTER.TXN field should be set to ALL

Note: - This will have an impact on the TXN.ENTRY enquiry and few other teller enquiries. They will not be able to view the Transaction entries for those applications if this field is set

### CONSOLIDATE.COND

PROFIT&LOSS record and ASSET&LIAB

Field UPDT.CONSOL.PROFIT to be set to NONE.

This will suppress the update to RE.CONSOL.PROFIT and also this will make sure when updating STMT.ENTRY, CATEG.ENTRY or RE.CONSOL.SPEC.ENTRY UNIQUE ids are used (based on port number) to avoid lock collisions.

### SPF

CACHE.EXPIRY is set to NULL
This is to make sure the session cache is kept throughout the life of the user session.

INFO.JOURNAL is set to null
This setting will suppress the update on F.JOURNAL file

CACHE.SIZE is set to 500
Specifies the cache size to be used within the T24 system for record writes.
Internally within the T24 system this field defines the size of the cache held before records are written to file. If the CACHE.SIZE is huge then it will affect the performance.

POS.MVMT.HIST is set to 1
This is will make sure that FBNK.POS.MVMT.HIST file will have only one days position records, which will help to increase the performance of EB.PRIME.FILES job

### OFS.SOURCE

All OFS.SOURCE records released in Model Bank should have OFS.REQUEST.DETAIL and OFS.LOG switched off. In order to achieve this, the values should be as follows –

LOG.LEVEL.DETAIL - EXCEPT
This will suppress the OFS detailed log instead log file will be updated only when there is an exception or error

MAINT.MSG.DETS - NULL
This will suppress the update on F.OFS.REQUEST.DETAIL file

### AUTO.ID.START

Explain in general terms for local devs....

Field UNIQUE.ID should be set to YES for the following applications –

```
UNIQUE.LIST           = 'FUNDS.TRANSFER'
UNIQUE.PREFIX.LIST    = 'FT'

UNIQUE.LIST<-1>       = 'TELLER'
UNIQUE.PREFIX.LIST<-1>  = 'TT'

UNIQUE.LIST<-1>       = 'SEC.TRADE'
UNIQUE.PREFIX.LIST<-1>  = 'SCTRSC'

UNIQUE.LIST<-1>       = 'SECURITY.TRANSFER'
UNIQUE.PREFIX.LIST<-1>  = 'SECTSC'

UNIQUE.LIST<-1>       = 'REPO'
UNIQUE.PREFIX.LIST<-1>  = 'RP'

UNIQUE.LIST<-1>       = 'SC.OBC.CUST.DETAIL'
UNIQUE.PREFIX.LIST<-1>  = 'SCOBCD' ;

UNIQUE.LIST<-1>       = 'SC.SOO.CUST.DETAIL'
UNIQUE.PREFIX.LIST<-1>  = 'SCSOCD' ;

UNIQUE.LIST<-1>       = 'SC.SEC.TRADE.CUST.DETAIL'
UNIQUE.PREFIX.LIST<-1>  = 'SCSTCD'
```

This is to make sure UNIQUE transaction ids are used for the above applications, by using UNIQUE ids lock collisions can be avoided and also scalability can be achieved across multiple T24 application servers

### PM.AC.PARAM – check for the functionality and amend accordingly

ACC.OR.CAL field modified to CAL for all accounts excepting NOSTROS

R07.004 MODEL BANK  PM.AC.PARAM SEE

POSITION.ID..... CAS  CASH FLOW

```
---------------------------------------------------------------------------
  1. 1 GB DESCRIPTION. PM RULES FOR ACCOUNTS
  2. 1 GB SHORT.NAME.. PM AC
  3 CALENDAR..... CAL              PMCALENDAR
  4 INTEREST.RATES.... NO
  5. 1 CATEGORY.FROM.. 1-000
  6. 1 CATEGORY.TO.... 4-999
  8. 1 CLASS.OVERNIGHT ACASC           AC CUST
 13. 1 ACC.OR.CAL..... CAL
 14. 1 GROUP.KEY...... 1
 15. 1. 1 CURRENCY.... DEFAULT
 16. 1. 1 BAL.SPD.GRP. 1
 17. 1. 1 INT.SPD.GRP. 1
  5. 2 CATEGORY.FROM.. 5-000
  6. 2 CATEGORY.TO.... 5-999
  8. 2 CLASS.OVERNIGHT ACBSC           AC NOSTRO
 13. 2 ACC.OR.CAL..... ACC
 15. 2. 1 CURRENCY.... DEFAULT
 16. 2. 1 BAL.SPD.GRP. 1
 17. 2. 1 INT.SPD.GRP. 1
  5. 3 CATEGORY.FROM.. 6-000
  6. 3 CATEGORY.TO.... 8-999
  8. 3 CLASS.OVERNIGHT ACASC           AC CUST
 13. 3 ACC.OR.CAL..... CAL
 15. 3. 1 CURRENCY.... DEFAULT
 16. 3. 1 BAL.SPD.GRP. 1
 17. 3. 1 INT.SPD.GRP. 1
  5. 4 CATEGORY.FROM.. 10-000
  6. 4 CATEGORY.TO.... 18-999
  8. 4 CLASS.OVERNIGHT ACASC           AC CUST
 13. 4 ACC.OR.CAL..... CAL
 14. 4 GROUP.KEY...... 1
 15. 4. 1 CURRENCY.... DEFAULT
16. 4. 1 BAL.SPD.GRP. 2
 17. 4. 1 INT.SPD.GRP. 2
 18. 1 SPD.GRP.SIGN... 1.BOTH
 19. 1. 1 PERIOD.ID... CAL
 20. 1. 1 PERCENTAGE.. 100.00
 18. 2 SPD.GRP.SIGN... 2.BOTH
 19. 2. 1 PERIOD.ID... 10Y
 20. 2. 1 PERCENTAGE.. 100.00
 21. 2. 1 PRD.INT.TYPE 5.00
 25 CATEGORY.POSITION. 5
```

By doing the above PM.EOD.ACC job time can be drastically reduced, instead of reading the account balance from the account table PM.EOD.ACC will get the information from CAL itself which in turn reduces the run time of the job.

### PM.POSN.CLASS

All the PM.POSN.CLASS ids for FT and TELLER needs to changed like below. This will stop the update on PM.DRILL.DOWN files for FT and TELLER type of transaction for local currency

PM.POSN.CLASS.ID.. FTASC

---

1 SHORT.NAME..... FT O/L CUS

2. 1 GB DESCRIPTION. FT ON-LINE CUSTOMER

3 PRODUCT..... FT

4. 1 SKP.TXN.REF.CCY USD block drill down for this currency usually local

5. 1 SKP.TXN.REF.PER 2 For today + 2 days

6 POSITION.TYPE..... CAS

7 ORIGINAL.CLASS.... ACASC AC CUSTOMER (ON-LINE + EOD)

10 CURR.NO..... 3

### AC.STMT.PARAMETER

```
   ID..............  SYSTEM
 ------------------------------------------------------------------------------
  1 IF.NO.MOVEMENT.... NO
  2 DESCRIPT.STATEMENT NO
  3 INT.CLOSING.ADVICE Y
  4 INTEREST.SCALE.... NO
  5 TAX.ADVICE........ NO
  6 MIN.MONTHS.STMT... 6
  7 FIRST.STMT........ NO
  8 OTHER.OFFICER..... Y
 19 CURR.NO........... 2
 20. 1 INPUTTER....... 24_TAABS.INPUTT___OFS_TAABS
 21. 1 DATE.TIME...... 16 APR   07 16:00
 22 AUTHORISER........ 24_TAABS.INPUTT_OFS_TAABS
 23 CO.CODE........... GB-001-0001        PZERO SYSTEM
 24 DEPT.CODE......... 1
```

In AC.STMT.PARAMETER fields INTEREST.SCALE and TAX.ADVICE needs to be set as NO

This is to reduce the COB time, since INT.STMT.AND.ADV report which will run for long time during COB if all accounts have INTEREST.SCALE set to Y

### EB.EXTRACT.PARAMETER

If there is a record ONLINE in EB.EXTRACT.PARAMETER then the routine attached inside the application needs to be reviewed. It's recommended not to use ONLINE extraction which will cause performance issues during all online transactions. It's recommended to use OFFLINE extraction, so care must be taken before creating ONLINE record in EB.EXTRACT.PARAMETER table

### DESKTOP.PARAMETER

It's recommended not to use REALTIMEENQUIRY to get optimum performance from T24, so set ATTRIBUTES field in DESKTOP.PARAMETER to null (it should not contain REALTIMEENQUIRY)
If REALTIMEENQUIRY attribute is set then each ONLINE transaction will do more IO which will cause performance degrade, its better to avoid using REALTIME enquiries in Desktop

```
   SYSTEM SPEC..... - SYSTEM
 ------------------------------------------------------------------------------
  1. 1 ATTRIBUTES.....
```

### FT.TXN.TYPE.CONDITION

```
   TRANSACTION.TYPE.. AC
 ------------------------------------------------------------------------------
  1. 1 GB DESCRIPTION. Account Transfer
  2. 1 GB SHORT.DESCR. Accnt Transfer
  3 TXN.CODE.CR..... 213              Transfer
  4 TXN.CODE.DR..... 213              Transfer
  5 STO.TXN.CODE.CR... 214              Standing Order
  6 STO.TXN.CODE.DR... 214              Standing Order
  7 DR.CHARGE.TXN.CODE 234              Account Transfer Charges
  8 DR.CHEQUE.TXN.CODE 201              Outward Cheque - Dr
 11 FORW.VALUE.MAXIMUM +05W
 12 BACK.VALUE.MAXIMUM -05W
 13. 1 PAYMENT.TYPE... ALL
 14. 1 PAYMENT.VALUE.. Y
 15. 1 CUSTOMER.FLOAT. 0
 16. 1 SAME.CUST.FLOAT 0
 17 DR.ADVICE.REQD.Y.N N
 18 CR.ADVICE.REQD.Y.N N
```

set Debit/Credit Advice to NO for "AC" transaction type.

Regarding other transaction types, banks normally would like to produce "Debit Advice" for Outward Telex/Draft/Cheque related transactions to intimate charges & exchange rate. Similarly for Inward Transaction types, they normally require a "Credit Advice" to intimate the client for incoming funds less charges, if any. So all the FT transaction types needs modification for CR and DR advice settings

### ACCOUNT.STATEMENT

```
  ACCOUNT.NUMBER.... USD-10001-0001      Cash
 ------------------------------------------------------------------------------
 31 PRINT.STMT..... NO
 32 PASSBOOK.ID.....
 33 CONS.SB.PASSBOOK..
 34 LAST.MT941.DATE...
 35 LAST.MT941.BAL....
 36 LAST.MT941.STMT.NO
```

For all internal and nostro accounts PRINT.STMT field in ACCOUNT.STATEMENT needs to be set to NO. This is to suppress account statement generation for internal and nostro accounts during COB, which will help to reduce the COB time drastically in high volume client sites

### ACCT.GROUP.CONDITION

```
  GROUP.CCY.DATE.... 1EUR            Current Account Personal Euro
 ------------------------------------------------------------------------------
 33 DEFER.DB.INT.DAYS.
 34 DB.INT.PENDING.CAT
 35 TAX.PENDING.CAT... 12013          Deffered Charges -Accounts
 36 CORR.AT.CAP.DATE..
 37. 1 TXN.THRESHOLD.. 10
 38. 1. 1 TXN.CODE.GRP 2              Cash Withdrawal
```

In ACCT.GROUP.CONDITION, TXN.THRESHOLD field needs to be null to avoid necessary IO during IC.COB. This needs to done for all ACCT.GROUP.CONDITION records

### ACCOUNT

Explain HVT and subaccounting

Formatted: Normal

It's recommended that all high volume internal and nostro accounts are configured with sub accounts

In ACCOUNT table field MAX.SUB.ACCOUNT should be set to ~~99~~ for all internal and nostro accounts like shown below

```
   ACCOUNT.NUMBER.... USD-10001-1501      USDTILL2
 ------------------------------------------------------------------------------
157 CREDIT.CHECK.....
 158 AVAILABLE.BAL.UPD.
```

159 CONSOLIDATE.ENT...
**160 MAX.SUB.ACCOUNT... ~~99~~999 -**

Also for clients who are prior to release R6 its also recommended to set CONSOLIDATE.ENT to DEFAULT for internal and nostro accounts, this will make sure consolidation of entries (STMT.ENTRY or CATEG.ENTRY) for all internal and nostro accounts

### AC.ENTRY.PARAM

IF OFS clearing is used, then certain fields in AC.ENTRY.PARAM needs to be setup accordingly.

Ex:

KEY..... CLEARING
---
7 CONTRA.ACCT.CATEG. 10001 Cash
8. 1 CONTRA.CCY.....
9. 1 CONTRA.ACCT....
12 ACCOUNTING.MODE... SAO
13 CONTRA.ENTRY..... CURRENCY.NET
31. 1 PRODUCE.REPORT.
33 UPDATE.REPORT.FILE

For High volume transactions, fields CONTRA.CCY and CONTRA.ACCT needs to be set to Null. This is done to avoid locking of Internal account records. Also field ACCOUNTING.MODE needs to be set to 'SAO', so that any overrides raised on accounting entries will not stop the processing of the OFS Message. Field CONTRA.ENTRY set to 'CURRENCY.NET', which will reduce the number of contra entries raised if the OFS clearing messages are bulked together. Fields PRODUCE.REPORT and UPDATE.REPORT.FILE needs to be set to null to avoid producing reports otherwise clearing transactions wont scale across multiple sessions. The other important setting for clearing transactions is to setup sub accounts for CONTRA.ACCT.CATEG internal accounts this way lock collisions on contra accounts can be drastically reduced in turn performance will increase.

### ~~USER~~

The logging of events to PROTOCOL needs to be stopped in case of USER Ids being used in OFS messages (GENERIC.USER in OFS.SOURCE or COB.USER in TSA.SERVICE).
This can be done by setting up the following fields to 'NO'. This will help reduce lots of I/O to file F.PROTOCOL.

USER.ID..... OFSUSER

25 SIGN.ON.OFF.LOG... NO
26 SECURITY.MGMT.L... NO
27 APPLICATION.LOG... NO
28 FUNCTION.ID.LOG... NO

**COB REPORTS TO BE TURNED OFF review and change ((check the OFS source attributes field),**

LI.COLLAT.SUM
LI.COLLAT.DET
LIMIT.CURRENCY.EXP
LIMIT.COMMODITY.EXP
LIMIT.COUNTRY.EXP
LIMIT.REP.GENERAL
RE.ACCT.TRIAL.BAL
TRANS.JOURNAL2
EOD.RE.STAT.PRINT>DAILY
RGS.BALANCE.LIST
jBASE review and change

### Environment variables

| Variable | Suggested Value | Description |
|---|---|---|
| | | |
| JBC_UNLOCK_LASTCLOSE | | |
| JBC_EXIT_ZEROREAD | 1 | there have been a couple of things added to jbase 4.0 to try and catch the runaway read errors (usually a desktop disconnect, resultining in read() returning -1 repeatedly). Unfortunately, these were done in Patch 4_00271 (4 March 2004), I think HNB are up to Patch 256.<br>exit if no timeout and zero read |
| JBC_EXIT_FAILEDREAD | 1 | exit if read failed for no reason |
| JBC_WARNLEVEL | 30 | |
| JEDI_NOSHMEM | 1 | Prevents update of shared memory details and makes forking a process much faster – i.e. doing a select. Only useful if jbase 4.0 is used |
| JBASEUNIQUE | JBASEUNIQUE=$HOME/workfile | Use JBASEUNIQUE, i.e. not JBCBASETMP |
| JEDI_XMLORACLE_DELETE_NULL | 1 | |
| JEDI_XMLORACLE_SORT_READ | 1 | |

| | | |
|---|---|---|
| JBASE_LOCK_INFO | 1 | |
| JBASE_ERRMSG_DIVIDE_BY_ZERO | 1 | |
| JBASE_ERRMSG_ZERO_USED | 1 | |

## Locking Mechanism and Parameters

Locks plays an important role in any OLTP application, in T24 locks are unavoidable. T24 uses hash file locks.

There are four rules of record locking. They are:

1. Always take a lock on any record you intend to update.
2. All locks must be implicitly or explicitly released.
3. A record should not be locked for longer than necessary.
4. Use caution when OPENing files.

The above rules must be followed to have a optimum application logic. So while analysing T24 performance problems lock performance also needs to be carefully studied. Based on clients requirement and architecture needs the best locking mechanism needs to be finalised.

### NFS locks

NFS locks are the old method of locking when have multiple T24 Application server, in this case bnk.data or /tmp/jbase or JBCLOCKDIR will be mounted centrally in an NFS server and shared across all available T24 application server

Some of the parameters to tune are given below

- On the NFS Client server
/usr/sbin/chnfs -n '15' -b '15' -l '15' '-B'

- Mount options on the Application server
dio,bg,soft,intr,sec=sys,rsize=32768,wsize=32768,timeo=600,rw

- On the NFS Server
/usr/sbin/chnfs -n '4' -b '10' -l '10' '-B'

- nfso changes for nfs_max_write_size and nfs_max_read_size to 8192

- also to diagnose the problem turn off nfs_device_specific_bufs

- also what does the nfsstat is giving diagnose the problem, turn off nfs_device_specific_bufs

### Distributed Lock Arbiter – JDLS

Although T24 has been successfully deployed over Multiple Applications Servers certain idiosyncrasies related to operating systems, performance and recovery of file locks over networked file systems have prompted the requirement to provide an alternative lock strategy for the T24 Multiple Application Server deployment. As such the jBASE Distributed lock service is now offered as an alternative mechanism, to the networked file system, by which locks can be propagate between servers.

The Distributed lock service can be deployed as a service executing on a dedicated server or, as is more likely, deployed on one or two of the Application Servers in the Multiple Application Server configuration.

The Distributed lock service is provided via a daemon process executing on Linux/Unix systems or as an installable service for the Windows platform. The Distributed lock service can be initiated using either the lower case form of the executable, 'jdls', or the jBASE capitalized convention for process daemons, jDLS, (jBASE Distributed Lock Service).

The Distributed lock service also supersedes the jRLA, (jBASE Record Lock Arbiter), which provided the shared memory lock mechanism for record locks. Linux/Unix system boot scripts should be modified to initialise the lock mechanism using the jDLS rather than the jRLA executable.

The connection comprises of a TCPIP socket, which uses a network byte oriented structure to pass specific lock only information. The information for example being lock type, client port number, client process id, client thread id, etc

*So the network connection between the jdls server and T24 server needs to be proper to have a better performance with JDLS locking mechanism. Also tcp/ip parameters like Tcp_send_buff_size, tcp_recv_buff_size, tcp_time_wait etc needs to be set to proper values in T24 application server(s). Also consider that the server in which jdls is running will have additional processes (each connection will create on jdls client process) running which might take more system resources*

There are two ways to deploy jDLS server

```
jDLS –ibs13000,50
```

This command initializes the jDLS to start both the jDLS Shared memory lock monitor service and the jDLS Distributed lock listener service in background. The shared memory will be configured to provide 13000 locks with 50 locks per group. The lock table algorithm will actually use the next prime number for the number of groups to provide a better spread of locks over the groups and so in this case configure the shared lock table with 13150 locks in 263 groups with 50 locks per group.

jDLS –ibD

This command instructs the jDLS to initialize ONLY the jDLS Distributed lock listener service. Unless the jDLS Shared memory lock monitor/arbiter service had been previously started then OS file locks will be used for the default lock mechanism for both distributed locks and local process locks. In this mechanism with multiple application server mode /tmp/jbase or JBCLOCKDIR directory needs to be in NFS/GPFS. So when using this mechanism NFS parameters need to be properly configured to have optimum performance.

For more information on jDLS please refer to jDLS user guide

### jRLA

jRLA is a component of jBASE which helps to control locks for T24. jRLA uses shared memory locking model, all locks will be held in shared memory which can be accessed by each T24/jBASE processes. jRLA gives lot of benefit over OS locks (internally jRLA also can use OS locks).

To get an optimum performance jRLA must be run with –ibs option, this will help to maintain the binary locks in OS locks mechanism and rest of the locks will be maintained in shared memory. One of the important criteria is to get an optimum record locks and group locks number for jRLA. Requirement for number of record locks and group locks changes per client/installation, so based on clients transaction volume optimum number of record locks and group locks needs to be finalised. To start with the following can be used

**jrla –ibs30700,100**

one of the main advantage of jRLA compared to OS locks is locks monitoring, using jrla locks can be monitored efficiently with jrla –dvL command

Note:- based on few performance tests it was noticed that jRLA locking mechanism works faster than the OS locks and also it scales well with increasing demand in lock requests.

Its also recommended that /tmp/jbase directory in unix is placed in a faster disk to get better performance with jRLA.

### OS Locks

jBASE supports OS locks perfectly, in case of OS locks Jbase allows the operating system to control the locks. OS lock performance is entirely dependent on the kernel level settings of the operating system in which T24/jBASE is running. OS locks should be first preference if client is running on a single server mode ie. T24/jBASE is running on is only one server. Please refer to operating system user manuals to get OS lock kernel settings

It's recommended that /tmp/jbase directory in unix is placed in a faster disk to get better performance with OS locks. When running in HP operating system check **nflocks** kernel parameter and tune it to the proper value to get optimum performance

#### Database lock

Formatted: Heading 4

Explain database locks.

#### File Distribution

For efficient I/O it is imperative that large files (>500mb) are distributed. This ensures that they do not reach the 2gb limit and if distributed sensibly then the data most frequently used will always be in memory. Even with the advent and 64-bit builds which remove the 2 Gb limit, it is still recommended to distribute large filed.
The STMT.ENTRY/CATEG.ENTRY/RE.CONSOL.SPEC.ENTRY files can redistributed so that the part files are populated by day. Create 31 part files with the following algorithm:

```
INT((ID[4,2]+0.5)/3.22+1)
```

As each STMT.ENTRY key begins with the system date in internal format this takes a day number 0-99 and converts it to 1-31. This means the entries for a specific day (generation day) will reside in one file and consequently this file should be loaded into memory.

Also do the following in .profile
Export JBASE_DISTRIB_FASTSCAN=1 to make the part file scanning faster

Note:- From R8 and jbase 5.0 there is no need to distribute the file, instead its recommended to create the file as JR type

Operating System
This section deals with the operating system settings that can be tuned. This tuning is used for tuning the OS to optimise performance for the T24 Server Sessions – i.e. where the T24 business logic is running. However, the principles can be applied to the operating system regardless of which logical tier is being used.

### Generic -– tba Include all the generic OS kernel parameters

Although the exact parameter names and values vary from operating system to operating system, and from machine to machine, in general T24 requires the following areas to be considered:

Permissions
Number of allowed processes
Number of allowed threads
Number of open files
Ulimit settings

### AIX - tba

### HP - tba

### Windows - tba

# ORACLE

### Database Version:
Minimum requirement is version 10.2.0.3

### Database Name:
This is client specific according to Oracle's guidelines i.e. it can be up to 8 characters. The following characters are valid in a database name: alphanumeric characters, underscore (_), number sign (#), and dollar sign ($). No other characters are valid.

### Database Character set :
AL32UTF8

### Database Block Size:
4k or 8k

### Tablespaces

All tablespaces should be locally managed with Automatic Segment Space Management (ASSM) and autoextensible where space permits.

**SYSTEM** tablespace
Recommend 1gb

**SYSAUX** tablespace
Recommend 500m, or more if AWR snapshots are taken more frequently than every half hour, or retention is longer than one week.

**TEMPORARY** tablespace
Recommend 1gb and adjust accordingly

**UNDO** tablespace
Size depends on the retention period. Recommend 1gb-3gb and adjust as required.

**T24 Tablespaces ***
One is required for application data, for example T24DATA, but the name is entirely client specific. The size depends upon the expected volume and estimation period.
One is required for application indexes, for example T24INDEX but name is entirely client specific. The size depends upon the expected volume and estimation period.

* Multiple tablespaces can be used if the T24 Mulit-Company module is selected and each Company has its own Oracle schema. Each schema can then have its own DATA and INDEX tablespaces if required.
In addition, once the database has been populated with the initial data setup, any tables can subsequently be moved to different tablespaces, if required.

### Redo Logs:

Redo logs should be sized so that they switch approximately every 20 minutes. The log switches are recorded in the Oracle alert_*sid*.log. It is recommended to start with 500m each and monitor and adjust accordingly.

### Database User:

Only one database user is required, typically called T24, but the name and password is client specific.

The default tablespace must be the application data tablespace, with quota unlimited required on the application data and index tablespaces. The Temporary tablespace is allocated automatically by default. No profiles are used other than the default. The user requires ownership of one pl/sql package (jbasexml) and one table (stubfiles) which are created during the initial data setup.

The user requires the following privileges/roles:
connect, resource, query rewrite, create synonym, drop any procedure, create any table, select any table, update any table, insert any table, delete any table, drop any table, create any index, alter any index, drop any index, create any view, drop any view, create any directory, select_catalog_role, grant xdbadmin and revoke unlimited tablespace.

### Oracle Parameters

The following Oracle parameters should be set. Most of the Oracle parameters are dynamic and can be changed with an alter system command, without having to restart the database.

optimizer_index_cost_adj=1
open_cursors=5000
session_cached_cursors=5000
cursor_sharing=similar
query_rewrite_enabled=TRUE
query_rewrite_integrity=TRUSTED
undo_management=AUTO
undo_retention=10800
log_checkpoint_timeout=0
sga_max_size=6G (depends on volumes adjust as necessary)
sga_target=4G (depends on volumes adjust as necessary)
aq_tm_processes=2
pga_aggregate_target=1g (adjust as necessary)
workarea_size_policy=AUTO

**For HP platforms only:**
hpux_sched_noage=154 (for HP/UX 11.0)
hpux_sched_noage=178 (for HP/UX 11i)

LOB Parameter Settings

There are two recommended performance settings for lobs CACHE and RETENTION.

### CACHE

For performance benefits it is recommended that the lob segments are set to CACHE
The CACHE storage parameter causes LOB data blocks to be read/written via the buffer cache. With the CACHE option, LOB data reads show up as wait event 'db file sequential read', writes are performed by the DBWR process.

In-line Lobs are not affected by the CACHE option as they reside with the other column data, which is typically accessed via the buffer cache. The CACHE option gives better read/write performance than the NOCACHE option. Currently, the CACHE option is not the default setting for the LOBS, so after the initial or subsequent population of the database, and when any new tables are created, a script needs to be run to correct the settings. The script can be found in Appendix 1

### RETENTION

Consistent Read (CR) on LOBs uses a different mechanism than that used for other data blocks in Oracle. Older versions of the LOB are retained in the LOB segment and CR is used on the LOB index to access these older versions (for in-line LOBs which are stored in the table segment, the regular UNDO mechanism is used). It is recommended that time-based retention using the RETENTION keyword is preferred; this specifies how long older versions are to be retained.

RETENTION is a keyword in the LOB column definition. No value can be specified for RETENTION. The RETENTION value is implicit. If a LOB is created with database compatibility set to 9.2.0.0 or higher, undo_management=TRUE and PCTVERSION is not explicitly specified, time-based retention is used. The LOB RETENTION value is always equal to the value of the UNDO_RETENTION database instance parameter.
The AUTO setting of the UNDO_MANAGEMENT parameter, does not AUTOMATICALLY manage the undo retention for lobs, so we have to manually increase the undo_retention parameter in most cases, to avoid getting the "snapshot to old errors". The UNDO tablespace should be configured to autotextend and the recommended initial UNDO_RETENTION setting is 10800 (3 hours). This should be monitored and increased if necessary. Some sites are running with 14400 (4 hours).

Due to a bug in Oracle: Bug 5176017 - Exp/imp loses "RETENTION" settings of LOB columns, the RETENTION setting for lobs is not retained when data is loaded via an export/import or data pump export/import, therefore after populating a database after an import, a script needs to be run to correct the settings. The script can be found in Appendix 2.

Listener, Tnsnames and SQLnet.ora Parameters

### Session Data Unit (SDU) Size

Before sending data across the network, Oracle Net buffers and encapsulates data into the session data unit (SDU). Oracle Net sends the data stored in this buffer when the buffer is full, flushed, or when database server tries to read data. When large amounts of data are being transmitted or when the message size is consistent, adjusting the size of the SDU buffers can improve performance, network utilization, or memory consumption.

The amount of data provided to Oracle Net to send at any one time can be referred to as the message size. Oracle Net assumes by default that the message size will normally vary between 0 and 2048 bytes, and infrequently, will be larger than 2048. If this assumption is true, then most of the time, the data will be sent using one SDU buffer (default size 2048). The SDU size can range from 512 bytes to 32767 bytes. It is possible to optimize data transfer by adjusting the size of the SDU.

The session data unit (SDU) and transport date unit (TDU) parameters are located in the *tnsnames.ora* and *listener.ora* files. SDU specifies the size of the packets to send over the network. Ideally, SDU should not surpass the size of the maximum transmission unit (MTU). MTU is a fixed value that depends on the actual network implementation used. Oracle recommends that SDU be set equal to MTU.

The TDU is the default packet size used within Oracle Net to group data together. The TDU parameter should ideally be a multiple of the SDU parameter. The default value for both SDU and TDU is 2,048, and the maximum value is 32,767 bytes.
It is recommended that these parameters are set to the maximum values of 32,767 and examples are given in Appendix 3 and 5.

### Persistent Buffer Flushing for TCP/IP

Under certain conditions for some applications using TCP/IP, Oracle Net packets may not get flushed immediately to the network. Most often, this behavior occurs when large amounts of data are streamed. The implementation of TCP/IP itself is the reason for the lack of flushing, causing unacceptable delays. To remedy this problem, specify no delays in the

buffer flushing process. This can be done by setting TCP.NODELAY in the sqlnet.ora file, see Appendix 4.

### Configuring i/o Buffer Space

The RECV_BUF_SIZE and SEND_BUF_SIZE parameters specify sizes of socket receive and send buffers, respectively, associated with an Oracle Net connection. To ensure the continuous flow of data and better utilization of network bandwidth, specify the I/O buffer space limit for receive and send operations of sessions with the RECV_BUF_SIZE and SEND_BUF_SIZE parameters.

For best performance, the size of the send and receive buffers should be set large enough to hold all of the data that may be sent concurrently on the network connection. Recommendations are given in Appendix 5.

In all cases for the above settings, refer initially to Operating system defaults.
Tracing can be enabled to investigate any issues, details of which can be found in the Oracle 10g Documentation.

Automatic Job Settings

From Oracle 10g onwards, Oracle configured two system jobs that run automatically in the pre-defined maintenance window from 10pm-2am. Therefore the following two jobs will run every night during this time period. If this is not convenient, they can be disabled and run manually, or re-scheduled for a more convenient time. The two jobs are as follows:

### Automatic Statistics Collection Job

The GATHER_STATS_JOB collects optimizer statistics for all objects in the database for which there are no statistics or only stale statistics. It is necessary to run this job on a regular basis to keep statistics up to date, although the frequency of the run depends upon the changing data volumes throughout the day/week.
The job can be disabled with the following command or via OEM.

**EXECUTE DBMS_SCHEDULER.DISABLE('GATHER_STATS_JOB');**

The job can be run manually from sqlplus, with the following command (where T24 is the name of the schema owner):

**EXECUTE**
**DBMS_STATS.GATHER_SCHEMA_STATS('T24',DBMS_STATS.AUTO_SAMPLE_SIZE);**

### Automatic Segment Advisor Job

The AUTO_SPACE_ADVISOR_JOB runs the Automatic Segment Advisor, which identifies segments that have space available for reclamation, and then makes recommendations that you can view with Enterprise Manager or a set of PL/SQL package procedures.
You can run the Segment Advisor manually to obtain more up-to-the-minute recommendations or to obtain recommendations on segments that the Automatic Segment Advisor did not examine for possible space reclamation.

The job can be disabled with the following command or via OEM.

**EXECUTE DBMS_SCHEDULER.DISABLE('AUTO_SPACE_ADVISOR_JOB');**

The job can be run manually via OEM or from sqlplus, by object or tablespace, using the dbms_advisor package. An example is given below for the ACCOUNT table:

```
Variable id number;
Begin
  declare
  name varchar2(100);
  descr varchar2(500);
  obj_id number;
```

```
 begin
 name:='T24_Account_Table';
 descr:='Segment Advisor Example';
 dbms_advisor.create_task (
   advisor_name     => 'Segment Advisor',
   task_id          => :id,
   task_name        => name,
   task_desc        => descr);
dbms_advisor.create_object (
   task_name        => name,
   object_type      => 'TABLE',
   attr1            => 'T24',
   attr2            => 'FBNK_ACCOUNT',
   attr3            => NULL,
   attr4            => NULL,
   attr5            => NULL,
   object_id        => obj_id);
 dbms_advisor.set_task_parameter(
   task_name        => name,
   parameter        => 'recommend_all',
   value            => 'TRUE');
 dbms_advisor.execute_task(name);
 end;
end;
/
```

## Creating Views

A create-view program is supplied with the XMLOracle driver and requires the information from the jBASE extended dictionary in order to create the view successfully. To generate this information, a standard selection rebuild for the table has to be initiated from the T24 application. For example, from the T24 command prompt type SS, I ACCOUNT (for the ACCOUNT table), set field 28 to Y and commit the change. This will populate the extended dictionary information, generate a create view script and run the create view script in the Oracle database to create the view.

The view, or subsequent views, can be run from the jsh prompt by typing the command explicitly for example:

create-view -v FBNK.ACCOUNT

For XMLOracle driver versions 2.x a view must be created in order for any function based indexes to work, as the index is based on the view.

For XMLOracle driver versions 3.x a view is not required to be created, as the function based indexes are not based on the view, but on XPATH expressions directly to the table, however the extended dictionary information is still required. Views can still be created on the tables

for easier readability of the data when viewing in the Oracle database directly, but are not required by the driver.

### Setting up Explain Plan in Oracle

The explain plan facility in Oracle is used to show the execution plan for a query and to see whether indexes are being used or not. Before it can be used, a script needs to be run to set up a PLAN table to capture the execution output.
Create the plan table by running the following script from sqlplus.

**@$ORACLE_HOME/rdbms/admin/utlxplan.sql**

Then test it by putting EXPLAIN PLAN FOR in front of the sql query you wish to test. Note that the query does not get executed, but a query optimization plan is generated, indicating the behaviour of the query were it to be executed.

For example:

**select count(*) from V_STFBNK_CUST000 where sector like '1%';**

Then run the following script:

**@$ORACLE_HOME/rdbms/admin/utlxpls.sql**

Your output should look something like this:

```
PLAN_TABLE_OUTPUT
--------------------------------------------------------------------------------
| Id  | Operation         | Name              | Rows  | Bytes | Cost  |
--------------------------------------------------------------------------------
|   0 | SELECT STATEMENT    |                   |    1 |    16 |    4 |
|   1 |  SORT AGGREGATE     |                   |    1 |    16 |      |
|* 2 |   INDEX RANGE SCAN   | IX_STFBNK_CUST000_C10 |  7584 |   118K|    4 |
--------------------------------------------------------------------------------
```

The plan shows that the query would utilize the function based index created for sector on the customer table.

### Creating and Testing Indexes

A program to create function based indexes is supplied with the XMLOracle drivers. The program is called create-extindex and is run form the jsh prompt. As explained the Creating Views above, the pre-requisites are that the extended dictionary information is populated, and for version 2.x drivers only, a view is created as well.

Documentation on index creation and any additional driver variables are included in the driver release notes supplied with each driver version. This section will go through an example of creating an index and illustrating how to test if the index is working successfully. The example shown is using a version 3 XMLOracle driver with Oracle 10.2.0.3.
Please also refer to the section above for setting up the PLAN table to use explain plan in Oracle.
Index creation should initially be tested in a QA or pre-production system, with sufficient data volumes to reflect the production database.
The Oracle parameter optimizer_index_cost_adj should be set to 1 to enable the function based indexes to be utilized.

### Example: Creating an index on the Category column of the Account table.

**jsh → create-extindex -v FBNK.ACCOUNT CATEGORY**

The –v (verbose) flag will display the sql script generated:

**CREATE INDEX nix_FBNK_ACCOUNT_c2 ON FBNK_ACCOUNT x (NVL(NUMCAST(extractValue(x.xmlrecord,'/row/c2')),0))**

The index will be created in the Oracle database and the index will be analyzed to update the Oracle stats.
The prefix nix shows that it is a numeric index, the Category column is stored as right justified. For left justified values a character index is created with an ix prefix. C2 is the column value stored in the xml string in the Oracle database..
Note: The index creation can take several minutes on large tables as the process is not parallelized. If this is a problem, it is best to capture the sql output from the screen, interrupt the process and run the command directly in the Oracle database via sqlplus, with the parallel option. The analyze process will need to be run as well as shown below:

**sql> CREATE INDEX nix_FBNK_ACCOUNT_c2 ON FBNK_ACCOUNT x (NVL(NUMCAST(extractValue(x.xmlrecord,'/row/c2')),0)) parallel nologging;**

**sql> exec dbms_stats.gather_index_stats('T24','NIX_FBNK_ACCOUNT_C2');**
(where T24 is the name of the schema owner in Oracle).

It is recommended that all index create scripts are saved, as they can then be used to populate additional environments in future.

## Testing the Index

Remove or clear the ORAdiriver.log, which is located in the bnk.run directory:

**jsh-->rm ORAdriver.log**

Temporarily set the debug variable to trap the sql in the ORAdriver.log:

**jsh-->export JEDI_XMLORACLE_DEBUG=1**

Run an enquiry on the account table, referencing the category column and clear the result set:

**jsh-->SELECT FBNK.ACCOUNT WITH CATEGORY EQ 1400**

16 Records selected

**>CLEARSELECT**
[528] List '0' cleared.

View the ORAdriver.log to see the sql generated by the XMLOracle driver:

**jsh-->vi ORAdriver.log**

3.2.3.156450 - 18946 - (select.b,7) - Thu Feb 14 17:16:29 - FBNK.ACCOUNT - FBNK_ACCOUNT - JQL: SELECT FBNK.ACCOUNT WITH CATEGORY EQ 1400
3.2.3.156450 - 18946 - (select.b,7) - Thu Feb 14 17:16:29 - FBNK.ACCOUNT - FBNK_ACCOUNT - SQL: SELECT t.RECID FROM FBNK_ACCOUNT t WHERE NVL(NUMCAST(EXTRACTVALUE(t.XMLRECORD,'/row/c2')),0)=1400

Copy the sql generated from the enquiry to sqlplus to run an explain plan. Add a semi-colon to terminate the statement:

**sql> explain plan for**
**SELECT t.RECID FROM FBNK_ACCOUNT t**
**WHERE NVL(NUMCAST(EXTRACTVALUE(t.XMLRECORD,'/row/c2')),0)=1400;**
Explained.

**SQL> @$ORACLE_HOME/rdbms/admin/utlxpls**

PLAN_TABLE_OUTPUT Plan hash value: 2765633205

| Id | Operation | Name | Rows | Bytes | Cost | (%CPU)| Time |
|---|---|---|---|---|---|---|---|
| 0 | SELECT STATEMENT | | 5012 | 9847K| 20 (0)| 00:00:01 |
| 1 | TABLE ACCESS BY INDEX ROWID| FBNK_ACCOUNT | 5012 | 9847K| 20 (0)| 00:00:01 |

```
|*  2 |   INDEX RANGE SCAN          | NIX_FBNK_ACCOUNT_C2 |  2005 |      |1   (0)| 00:00:01 |
--------------------------------------------------------------------------------
Predicate Information (identified by operation id):
---------------------------------------------------
2access(NVL("T24"."NUMCAST"(EXTRACTVALUE(SYS_MAKEXML("SYS_NC00003$"),'/row/c2'
)),0)=1400)
```

From the above PLAN output, it can be seen that the enquiry would use the index NIX_FBNK_ACCOUNT_C2 created on the Account table.

In addition to this, you can set the timing on to measure the time taken at the jBASE runtime level, to execute the enquiry, and also at the Oracle level. You can measure the times taken with and without the index created to see the effect on performance. Examples are given below:

**jsh -->time SELECT FBNK.ACCOUNT WITH CATEGORY EQ 1400**

16 Records selected

**usr: 0.00 sys: 0.00 elapsed: 0m0.02s**

```
SQL> set timing on
SQL> SELECT t.RECID FROM FBNK_ACCOUNT t
  2  WHERE NVL(NUMCAST(EXTRACTVALUE(t.XMLRECORD,'/row/c2')),0)=1400;

RECID
-----
16497
....
....
16427

16 rows selected.

Elapsed: 00:00:00.01
```

**Note:** You can also view the explain plan output in the ORAdriver.log by setting the environment variable **JEDI_XMLORACLE_EXPLAIN=1**, if you do not wish to use Oracle directly, for example:

**jsh→ export JEDI_XMLORACLE_EXPLAIN=1**
**jsh→ SELECT FBNK.ACCOUNT WITH CATEGORY EQ 1400**
**jsh-->vi ORAdriver.log**

3.2.3.156450 - 4356 - (select.b,7) - Thu Feb 14 17:32:07 - FBNK.ACCOUNT - FBNK_A

CCOUNT - explain: SELECT t.RECID FROM FBNK_ACCOUNT t WHERE NVL(NUMCAST(EXTRACTVA
LUE(t.XMLRECORD,'/row/c2')),0)=1400

3.2.3.156450 - 4356 - (select.b,7) - Thu Feb 14 17:32:07 - SQL Explain Plan: Pla
n hash value: 2765633205ÿ ÿ----------
----------ÿ| Id | Operation
| Name | Rows | Bytes | Cost (%CPU)| Time |ÿ----------
---
------ÿ| 0 | SELECT STATEMENT | | 5012 | 9847
K| 20 (0)| 00:00:01 |ÿ| 1 | TABLE ACCESS BY INDEX ROWID| FBNK_ACCOUNT
| 5012 | 9847K| 20 (0)| 00:00:01 |ÿ|* 2 | INDEX RANGE SCAN
| NIX_FBNK_ACCOUNT_C2 | 2005 | | 1 (0)| 00:00:01 |ÿ----------
---
------ÿ ÿQuery Block Name / Object Alias (identified by operation id):ÿ---------
----------ÿ ÿ 1 - SEL$1 / T@SEL$1ÿ
2 - SEL$1 / T@SEL$1ÿ ÿPredicate Information (identified by operation id):ÿ-----
----------ÿ ÿ 2 - access(NVL("T24"."NUMCAS
T"(EXTRACTVALUE(SYS_MAKEXML("SYS_NC00003$"),'/row/c2')),0)=1ÿ 400)ÿ
ÿColumn Projection Information (identified by operation id):ÿ----------
----------ÿ ÿ 1 - "T"."RECID"[VARCHAR2,255]ÿ
2 - "T".ROWID[ROWID,10],
NVL("T24"."NUMCAST"(EXTRACTVALUE(SYS_MAKEXML("SYS_NC000
03$"),'/ÿ row/c2')),0)[NUMBER,22]

**The variables JEDI_XMLORACLE_DEBUG and JEDI_XMLORACLE_EXPLAIN should only be set temporarily whilst index testing and should not be set in a production environment.**

## Troubleshooting:

**If you get errors creating an index it could be for one of the following reasons:**

Standard Selection has not been re-built for the table.
The column is defined as multi-value (only single value columns can be indexed unless the structured storage option is used)
The column contains an ITYPE.

**If the index gets created successfully, but does not get used:**

The selectivity of the values may be such that the majority of records are to be selected in an enquiry, in these cases Oracle may choose to do a table scan as it is deemed more performant.
The table is very small, and a table scan will be more performant.
The combination of expressions in the where clause may be too complex for the driver to pass to Oracle.
The Oracle parameter index_cost_adj=1 is not set in the database.

Please report any problems to the Temenos Help Desk and copy me (jbennett@temenos.com) for investigation.

Oracle Bugs Summary to be reviewed and include the latest bugs

The following is a summary of Oracle bugs experienced in Oracle version 10.2.0.3, either by ourselves of our clients. Full details of each bug can be found in the appendices.
Points to note are that all bugs are claimed to be fixed in version Oracle version 10.2.0.4, along with several others, so if you are not experiencing severe problems, it is best to wait for the 10.2.0.4 patch. Unfortunately we have no date for this yet, but it should be available in the next couple of months.
Interim patches or workarounds are available for the bugs listed below. Before applying interim patches it is recommended that the issue is raised with Oracle Support (TAR) giving details of the error and referencing the Bug number. Oracle will then request opatch inventory details to ensure the application of the patch is supported.

### Bug 4516865 Wrong permissions Set at Oracle 10g Installation

This has been an issue since the release of 10gR2 and there is a patch or workaround provided. The workaround involves resetting permissions at the UNIX level and is easy to apply. See Appendix 6 for details.

### Bug 6376915 High Watermark Enqueue Contention for ASSM LOB segments

This bug was hit when running very high volume inserts during a benchmark test and severely affected performance. The patch was applied which did make some improvement. The application of the patch also requires an event to be set as follows:
alter system set event = '44951 trace name context forever, level 1024' scope=spfile;
and requires the database to be re-started.

If your system has a high volume throughput, it is likely that this bug may be encountered. It can be seen by examining the statspack or AWR reports and the enq:HW will appear in the top 5 wait events.
If this is occurring, and performance is suffering, then request the patch, otherwise wait for 10.2.0.4. See Appendix 7 for details.

### Bug 5636728 LOB Corruption After LOB Shrink Operation

This bug was encountered by one of our clients when running the online segment shrink process to reclaim fragmented space. A patch is available for this and was applied and tested by our client. To date the patch has not fixed the issue, so it is not recommended to use this process for re-organisation, use the alter table move process instead. See Appendix 8 for details.

### Bug 5387030 Automatic Undo Retention Tuning Causes Extra Space Allocation

This bug was reported by one of our clients. It does not give an error but causes an overhead on storage. A work around is offered for this:
alter system set "_smu_debug_mode" = 33554432;
It was suggested that the client test the workaround before applying the patch fix. No feedback is available for this at the time of writing. See Appendix 9 for details.

### Bug 5442919 Expired Extents Not Being Reused

Again this bug was reported by one of our clients but we have not seen it during benchmark tests. This behaviour causes an error to be generated. If you are experiencing these errors frequently and it is affecting the system adversely, then I would suggest testing and applying the interim patch. No feedback is available for this at the time of writing. See Appendix 10 for details.

## Appendix 1 Lob Cache Script

```
REM Code to generate LOB Cache Script
REM This script should be run as the T24 schema owner.
REM
```

```
set head off echo off feed off ver off
set pagesize 50000
spool cache_lobs.sql

select 'alter table '||table_name||' modify lob ('||column_name||')(CACHE);'
from user_lobs;

spool off
set head on echo on feed on ver on

REM ************************************************
REM

REM Then run the script: cache_lobs.sql
REM
```

## Appendix 2 Lob Retention Script

```
REM Code to generate LOB Retention Script
REM This script should be run as the T24 schema owner.
REM

set head off echo off feed off ver off
set pagesize 50000
spool lob_retention.sql

select 'alter table '||table_name||' modify lob ('||column_name||')(RETENTION);'
from user_lobs;

spool off
set head on echo on feed on ver on

REM ************************************************
REM

REM Then run the script: lob_retention.sql
REM
```

## Appendix 3 tnsnames.ora File

Formatted: Indent: Left: 0 cm, First line: 0 cm, Space After: 0 pt

Check no –a|grep space

```
$ no -a|grep space
        tcp_recvspace = 262144
        tcp_sendspace = 262144
        udp_recvspace = 655360
        udp_sendspace = 65536
$
```

If its already high value then no need to setup specific BUFFER SIZE for tcp socket

```
# TNSNAMES.ORA  Sample Network Configuration File
#

MBDEMO =
  (DESCRIPTION =
     (SDU=32767)
    (ADDRESS_LIST =
     (ADDRESS = (PROTOCOL = TCP)(HOST = 11.11.1.111)(PORT = 1521))
     (SEND_BUF_SIZE=32767)
     (RECV_BUF_SIZE=32767)
   )
   (CONNECT_DATA =
     (SERVER = DEDICATED)
     (SERVICE_NAME = MBDEMO)
  )
 )
```

## Appendix 4 sqlnet.ora File

Formatted: Indent: Left: 0 cm, First line: 0 cm, Space After: 0 pt

```
# SQLNET.ORA Sample Network Configuration File
#

NAMES.DIRECTORY_PATH= (TNSNAMES, ONAMES, HOSTNAME)
TCP.NODELAY=YES

DEFAULT_SDU_SIZE=32767

RECV_BUF_SIZE=32767
SEND_BUF_SIZE=32767
```

```
# TRACE_LEVEL_SERVER=16
# TRACE_TIMESTAMP_SERVER
```

## Appendix 5 listener.ora File

Formatted: Indent: Left: 0 cm, First line: 0 cm, Space After: 0 pt

```
# LISTENER.ORA Sample Network Configuration File
#

LISTENER =
 (DESCRIPTION_LIST =
  (DESCRIPTION =
   (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC))
   )
   (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST =11.11.1.1)(PORT = 1521))
    (SEND_BUF_SIZE=32767)
    (RECV_BUF_SIZE=32767)
   )
  )
 )
SID_LIST_LISTENER =
 (SID_LIST =
  (SID_DESC =
   (SDU=32767)
   (TDU=32767)
   (MTU=32767)
   (SEND_BUF_SIZE=32767)
   (RECV_BUF_SIZE=32767)
    (GLOBAL_NAME = MBDEMO)
    (ORACLE_HOME = /oracle/10gR2)
    (SID_NAME = MBDEMO)
  )
 )
)

# TRACE_LEVEL_LISTENER=16
# TRACE_TIMESTAMP_LISTENER
```

## Appendix 6 Oracle Bug 4516865 Wrong permissions Set at Oracle 10g Installation

**Subject: SP2-0750 and "Message File Sp1.Msb Not Found" when Execute Sqlplus v10.2 as non-Oracle User on UNIX**

Doc ID: Note:356850.1 Type: PROBLEM
Last Revision Date: 30-AUG-2007 Status: PUBLISHED
**Applies to:**
SQL*Plus - Version: 10.2.0.1 to 10.2.0.3
Solaris Operating System (SPARC 32-bit)
All UNIX platforms Symptoms
New installation of Oracle Database 10g Release 2 (10.2.0.1).

When execute 'sqlplus' as a user outside of the dba or Oracle group, you get the following errors.
Error 6 initializing SQL*Plus
Message file sp1<lang>.msb not found
SP2-0750: You may need to set ORACLE_HOME to your Oracle software directory

**Syntax:**
$ sqlplus username/password
This problem did not occur in previous releases of Oracle.
Similar problems occur with other client tools, like IMP, EXP, etc.
**Changes**
New installation of Oracle Database 10g Release 2.
**Cause**
The files that sqlplus needs to execute do not have read/execute permissions on the O/S level.
**The source of the problem is base bug:**
**Bug 4516865 - Wrong Permissions After Installation In OH And Subsequent Directories**

For the client installation, which includes SQL*Plus, there are port specific bugs, some of which are:
Bug 4747264 - Wrong Permissions After Client Install Of 10gr2 (10.2.0.1) on LINUX
Bug 4932433 - Wrong Permissions After Client Install Of 10gr2 On Aix 5l (10.2.0.1).
Bug 4965571 - Wrong Permissions After Client Install Of 10gr2 (10.2.0.1) On Solaris

Wrong permissions are set on some files following installation of the 10gR2 base release (10.2.0.1).
Some directories under $ORACLE_HOME have no world read nor execute permission following the install (e.g. rwxr-x---).
This can stop local users not in the oracle code owner group from connecting using the bequeath driver.
**Solution**
There is a one-off patch for the base bug 4516865, but this fixes the permissions problem only on the server side. Only for Linux, this patch includes fix for both server and client side (Bug 4747264). Currently, there is still no patch for the client side for platforms other than Linux. However, there are a couple of workarounds:

I. Logged in as the Oracle user (or the user that installed the 10gR2 software), manually change the permissions on the client. For example:

chmod -R 755 <client_home>

In our case:

chmod -R 755 $ORACLE_HOME/sqlplus

II. If doing a recursive permissions command is not acceptable, then you will need to pinpoint exactly what files the client is reading at the time of execution, and manually change permissions only on those files. In our case, we need to pinpoint what files are being accessed by SQL*Plus. To implement this workaround, please execute the following steps:

1. As the non-Oracle user, run the truss utility to find out which files are being accessed. Sample command:
truss -aefo /tmp/truss_sqlplus.out sqlplus username/password

2. Use this truss_sqlplus.out trace file to see what files have error "EACCES" when attempting to access. In our case, the truss_sqlplus.out showed a problem accessing the following file:
$ORACLE_HOME/sqlplus/mesg/sp1us.msb
Another possible error to search for in the truss output is ENOENT. For example:
9775: open("./sqlplus/mesg/sp1us.msb", O_RDONLY) Err#2 ENOENT

3. Logged in as the Oracle user, change permissions on folders leading up to, and including sp1us.msb:
chmod 755 $ORACLE_HOME/sqlplus
chmod 755 $ORACLE_HOME/sqlplus/mesg
chmod 755 $ORACLE_HOME/sqlplus/mesg/sp1us.msb

4. After making above permission changes, a different error may appear when executing sqlplus as non-Oracle user, such as:

$ sqlplus username/password
SP2-1503: Unable to initialize Oracle call interface
SP2-0152: ORACLE may not be functioning properly

5. At this point, you need to re-run the truss (as non-Oracle) to see what other files are trying to be accessed. In our case, the following files were trying to get accessed, but showed "EACCES" failure:
$ORACLE_HOME/nls/data/lx1boot.nlb
$ORACLE_HOME/oracore/zoneinfo/timezlrg.dat

6. Logged in as the Oracle user, change permissions on these files and the directories leading up to these files.
chmod 755 $ORACLE_HOME/nls
chmod 755 $ORACLE_HOME/nls/data

chmod 755 $ORACLE_HOME/nls/data/lx1boot.nlb
chmod 755 $ORACLE_HOME/oracore
chmod 755 $ORACLE_HOME/oracore/zoneinfo
chmod 755 $ORACLE_HOME/oracore/zoneinfo/timezlrg.dat

7. Now, invoking sqlplus as a non-Oracle user was successful in our case.

**References**

Bug 4516865 - WRONG PERMISSIONS AFTER INSTALLATION IN OH AND SUBSEQUENT DIRECTORIES
Bug 4747264 - WRONG PERMISSIONS AFTER CLIENT INSTALL OF 10GR2 (10.2.0.1)
Bug 4932433 - WRONG PERMISSIONS AFTER CLIENT INSTALL OF 10GR2 ON AIX 5L (10.2.0.1).
Bug 4965571 - WRONG PERMISSIONS AFTER CLIENT INSTALL OF 10GR2 (10.2.0.1) ON SOLARIS
Note 110888.1 - How to Trace Unix System Calls
Note 365530.1 - Permissions not set correctly after 10gR2 installation

## Appendix 7 Oracle Bug 6376915 High Watermark Enqueue Contention for ASSM LOB segments

Subject: **Bug 6376915 - HW enqueue contention for ASSM LOB segments**

| | | | |
|---|---|---|---|
| Doc ID: | **Note:6376915.8** | Type: | **PATCH** |
| Last Revision Date: | **21-DEC-2007** | Status: | **PUBLISHED** |

Bug 6376915 HW enqueue contention for ASSM LOB segments
This note gives a brief overview of bug 6376915.

**Affects:**

| | |
|---|---|
| Product (Component) | Oracle Server (Rdbms) |
| Range of versions believed to be affected | Versions < 11 |
| Versions confirmed as being affected | 10.2.0.3 |
| Platforms affected | Generic (all / most platforms affected) |

**Fixed:**

| | |
|---|---|
| This issue is fixed in | 10.2.0.4 (Server Patch Set)<br>11.1.0.6 (Base Release) |

Symptoms:

Performance Of Certain Operations Affected
Waits for "enq: HW - contention"

Related To:

Datatypes (LOBs/CLOB/BLOB/BFILE)
ASSM Space Management (Bitmap Managed Segments)

**Description**

HW enqueue contention can occur for LOB segments which are ASSM managed as space allocation only acquires one block at a time.

With this fix ASSM lobs get a minimum number of chunks based on the value of event 44951 (up to 1024) which should help remove enqueue contention.

The full bug text (if published) can be seen at Bug 6376915 *(This link will not work for UNPUBLISHED bugs)*
You can search for any interim patches for this bug here Patch 6376915 *(This link will Error if no interim patches exist)*

## Appendix 8 Oracle Bug 5636728 LOB Corruption After LOB Shrink Operation

Formatted: Indent: Left: 0 cm, First line: 0 cm

Subject: **Bug 5636728 - LOB corruption / ORA-1555 when reading LOBs after a SHRINK operation**

| | | | |
|---|---|---|---|
| Doc ID: | **Note:5636728.8** | Type: | **PATCH** |
| Last Revision Date: | **21-DEC-2007** | Status: | **PUBLISHED** |

Bug 5636728 LOB corruption / ORA-1555 when reading LOBs after a SHRINK operation
This note gives a brief overview of bug 5636728.

**Affects:**

| | |
|---|---|
| Product (Component) | Oracle Server (Rdbms) |
| Range of versions believed to be affected | Versions < 11 |
| Versions confirmed as being affected | 10.2.0.3 |
| Platforms affected | Generic (all / most platforms affected) |

**Fixed:**

| | |
|---|---|
| This issue is fixed in | 10.2.0.4 (Server Patch Set)<br>11.1.0.6 (Base Release) |

| Symptoms: | Related To: |
|---|---|
| Error May Occur<br>Corruption (Physical)<br>ORA-1555 / ORA-22924 | Datatypes (LOBs/CLOB/BLOB/BFILE) |

**Description**
After shrinking an ENABLE STORAGE IN ROW LOB column selecting the LOB may fail with ORA-1555 / ORA-22924 errors .

**Workaround:**
Avoid using SHRINK on tables with LOB columns

The full bug text (if published) can be seen at Bug 5636728 *(This link will not work for UNPUBLISHED bugs)*
You can search for any interim patches for this bug here Patch 5636728 *(This link will Error if no interim patches exist)*

## Appendix 9 Oracle Bug 5387030 Automatic Undo Retention Tuning Causes Extra Space Allocation

Formatted: Indent: Left: 0 cm, First line: 0 cm

Bug 5387030 Automatic tuning of undo_retention causes unusual extra space allocation
This note gives a brief overview of bug 5387030.

**Affects:**

| | |
|---|---|
| Product (*Component*) | Oracle Server (*Rdbms*) |
| Range of versions *believed* to be affected | Versions >= 10.2.0.1 but < 11 |
| Versions *confirmed* as being affected | 10.2.0.1<br>10.2.0.2 |

| | |
|---|---|
| | 10.2.0.3 |
| Platforms affected | Generic (all / most platforms affected) |

**Fixed:**

| | |
|---|---|
| This issue is fixed in | 10.2.0.4 (Server Patch Set)<br>11.1.0.6 (Base Release) |

Symptoms:

Storage Space Usage Affected

Related To:
System Managed Undo (SMU)
V$UNDOSTAT
DBA_ALERT_HISTORY
_SMU_DEBUG_MODE

**Description**
When undo tablespace is using NON-AUTOEXTEND datafiles, V$UNDOSTAT.TUNED_UNDORETENTION may be calculated too high preventing undo block from being expired and reused. In extreme cases the undo tablespace could be filled to capacity by these unexpired blocks.
An alert may be posted on DBA_ALERT_HISTORY that advises to increase the space when it is not really necessary if this fix is applied. If the user sets their own alert thresholds for undo tablespaces the bug may prevent alerts from being produced.
**Workaround:**
alter system set "_smu_debug_mode" = 33554432;
This causes the v$undostat.tuned_undoretention to be calculated as the maximum of:
maxquerylen secs + 300
undo_retention specified in init.ora

## Appendix 10 Oracle Bug 5442919 Expired Extents Not Being Reused

**Formatted:** Indent: Left: 0 cm, First line: 0 cm

Bug 5442919 Expired extents not being reused (ORA-30036)
This note gives a brief overview of bug 5442919.
**Affects:**

| | |
|---|---|
| Product (*Component*) | Oracle Server (*Rdbms*) |
| Range of versions *believed* to be affected | Versions < 11 |
| Versions *confirmed* as being affected | 9.2.0.7<br>9.2.0.8<br>10.1.0.5<br>10.2.0.3 |
| Platforms affected | Generic (all / most platforms affected) |

**Fixed:**

| | |
|---|---|
| This issue is fixed in | 10.2.0.4 (Server Patch Set)<br>11.1.0.6 (Base Release) |

Symptoms:
Error May Occur
Storage Space Usage Affected
ORA-30036

Related To:
System Managed Undo (SMU)

**Description**
With plenty of expired space in the auto-managed undo tablespace (and many extents per offlined undo segment) SQL statements (DML) may fail with this error:
ORA-30036: unable to extend segment by <n> in undo tablespace '<ts>'
This message is also repeated for the same undo segment in the alert log:
"Failure to extend rollback segment <us#>"

```
{dbp5:oracle}/oracle/network/admin> mount
  node      mounted       mounted over    vfs      date       options
-------- ---------------  ---------------  ------ ------------ ---------------
       /dev/hd4       /             jfs2   Feb 21 02:16 rw,log=/dev/hd8
       /dev/hd2       /usr          jfs2   Feb 21 02:16 rw,log=/dev/hd8
       /dev/hd9var    /var          jfs2   Feb 21 02:16 rw,log=/dev/hd8
       /dev/hd3       /tmp          jfs2   Feb 21 02:16 rw,log=/dev/hd8
       /dev/hd1       /home         jfs2   Feb 21 02:16 rw,log=/dev/hd8
       /dev/hd11admin  /admin       jfs2   Feb 21 02:16 rw,log=/dev/hd8
       /proc          /proc         procfs Feb 21 02:16 rw
       /dev/hd10opt   /opt          jfs2   Feb 21 02:16 rw,log=/dev/hd8
       /dev/oraclelv  /oracle       jfs2   Feb 21 02:16 rw,log=/dev/hd8
       /dev/datalv    /oradata      jfs2   Feb 25 19:58 rw,cio,log=/dev/loglv00
```

**Note:** Make sure oracle file system is created as jfs2 and cio (concurrent io which will bypass OS file caching and use only oracle caching, this will allow oracle to have more memory)

DB2 – tba
Check with Gowri

SQL Server

Check with Charles - lee

# WEB SERVER

## Checking hardware configuration and settings – common for all web servers

An optimal hardware configuration enables applications to get the greatest benefit from performance tuning. The hardware speed impacts all types of applications and is critical to overall performance.

The following parameters include considerations for selecting and configuring the hardware on which the application servers run.

- **Optimize disk speed**
    - **Description:** Disk speed and configuration have a dramatic effect on the performance of application servers running applications that are heavily dependent on the database support, using extensive messaging, or processing workflow. The disk input or output subsystems that are optimized for performance, for example Redundant Array of Independent Disks (RAID) array, high-speed drives, and dedicated caches, are essential components for optimum application server performance in these environments.

      Application servers with fewer disk requirements can benefit from a mirrored disk drive configuration that improves reliability and has good performance.
    - **Recommendation:** Spread the disk processing across as many disks as possible to avoid contention issues that typically occur with 1- or 2-disk systems. Placing the database tables on disks that are separate from the disks that are used for the database log files reduces disk contention and improve throughput. With T24 Browser limited amount of information only written on web server level like log files, session sharing information (if cluster is used). So disk requirements are not too high for T24 Browser but if heavy logging is required then its better to place the log files under faster disks to get optimum performance.
- **Increase processor speed and processor cache**
    - **Description:** In the absence of other bottlenecks, increasing the processor speed often helps throughput and response times. A processor with a larger

L2 or L3 cache yields higher throughput, even if the processor speed is the same as a CPU with a smaller L2 or L3 cache.

- **Increase system memory**
  - **Description:** Increase memory to prevent the system from paging memory to the disk to improve performance. Allow a minimum of 256 MB of memory for each processor and 512 MB per application server. Adjust the available memory when the system pages and the processor utilization is low because of the paging. The memory access speed might depend on the number and placement of the memory modules. Check the hardware manual to make sure that your configuration is optimal.
  - **Recommendation:** For T24 Browser its recommended to have 2GB of memory per processor to have optimum throughput.
- **Run network cards and network switches at full duplex**
  - **Description:** Run network cards and network switches at full duplex and use the highest supported speed. . If we're tuning duplex speed manually, care should be taken to set full duplex on both NIC and Switches. Failure to do so would result in duplex mismatch (refer the auto negotiation table for details http://www.cisco.com/warp/public/473/46.html )
  - Full duplex is much faster than half duplex. Verify that the network speed of adapters, cables, switches, and other devices can accommodate the required throughput.
  - **Recommendation** Make sure that the highest speed is in use on 10/100/1000 MB Ethernet networks. Its also recommended to tune tcp/ip settings on kernel level.

    tcp_fin_timeout – keep this as less as possible, recommended 30secs
    netdev_max_backlog – recommended value 3000 or above
    somaxconn – recommended value 3000 or above

## Apache Tomcat

The list of parameters which needs to be tunes are given below

| | |
|---|---|
| enableLookups | Set to true if you want calls to request.getRemoteHost() to perform DNS lookups in order to return the actual host name of the remote client. Set to false to skip the DNS lookup and return the IP address in String form |

| | |
|---|---|
| | instead (thereby improving performance). By default, DNS lookups are enabled. |

| | |
|---|---|
| tcpNoDelay | If set to true, the TCP_NO_DELAY option will be set on the server socket, which improves performance under most circumstances. This is set to true by default. |

| | |
|---|---|
| processCache | (int)The protocol handler caches Http11NioProcessor objects to speed up performance. This setting dictates how many of these objects get cached. -1 means unlimited, default is 200. Set this value somewhere close to your maxThreads value. |

| | |
|---|---|
| maxThreads | The maximum number of request processing threads to be created by this **Connector**, which therefore determines the maximum number of simultaneous requests that can be handled. If not specified, this attribute is set to 40. If an executor is associated with this connector, this attribute is ignored as the connector will execute tasks using the executor rather than an internal thread pool. |

| | |
|---|---|
| connectionTimeout | The number of milliseconds this **Connector** will wait, after accepting a connection, for the request URI line to be presented. The default value is 60000 (i.e. 60 seconds). |

| | |
|---|---|
| maxSpareThreads | The maximum number of unused request processing threads that will be allowed to exist until the thread pool starts stopping the unnecessary threads. The default value is 50. |

| | |
|---|---|
| minSpareThreads | The number of request processing threads that will be created when this **Connector** is first started. The connector will also make sure it has the specified number of idle processing threads available. This attribute should be set to a value smaller than that set for maxThreads. The default value is 4. |

## IBM websphere

### Servlet caching

#### Procedure

1. In the administrative console, click **Servers** > **Application servers >** *server_name* **> Web container settings > Web container** in the console navigation tree.
2. Select **Enable servlet caching** under the Configuration tab.
3. Click **Apply** or **OK**.
4. Restart Web Sphere Application Server

### Tuning TCP/IP buffer sizes

WebSphere Application Server uses the TCP/IP sockets communication mechanism extensively. For a TCP/IP socket connection, the send and receive buffer sizes define the receive window. The receive window specifies the amount of data that can be sent and not received before the send is interrupted. If too much data is sent, it overruns the buffer and interrupts the transfer. The mechanism that controls data transfer interruptions is referred to as flow control. If the receive window size for TCP/IP buffers is too small, the receive window buffer is frequently overrun, and the flow control mechanism stops the data transfer until the receive buffer is empty.

#### About this task

Flow control can consume a significant amount of CPU time and result in additional network latency as a result of data transfer interruptions. It is recommended that you increase buffer sizes avoid flow control under normal operating conditions. A larger buffer size reduces the potential for flow control to occur, and results in improved CPU utilization. However, a large buffer size can have a negative effect on performance in some cases. If the TCP/IP buffers are too large and applications are not processing data fast enough, paging can increase. The goal is to specify a value large enough to avoid flow control, but not so large that the buffer accumulates more data than the system can process.

The default buffer size is 8 KB. The maximum size is 8 MB (8096 KB). The optimal buffer size depends on several network environment factors including types of switches and systems, acknowledgment timing, error rates and network topology, memory size, and data transfer size. When data transfer size is extremely large, you might want to set the buffer sizes up to the maximum value to improve throughput, reduce the occurrence of flow control, and reduce CPU cost.

Buffer sizes for the socket connections between the Web server and WebSphere Application Server are set at 64KB. In most cases this value is adequate.

Flow control can be an issue when an application uses either the IBM Developer Kit for Java(TM) JDBC driver or the IBM Toolbox for Java JDBC driver to access a remote database. If the data transfers are large, flow control can consume a large amount of CPU time. If you use the IBM Toolbox for Java JDBC driver, you can use custom properties to configure the buffer sizes for each data source. It is recommended that you specify large buffer sizes, for example,1 MB.

Some system-wide settings can override the default 8 KB buffer size for sockets. With some applications, for example, WebSphere Commerce Suite, a buffer size of 180 KB reduces flow control and typically does not adversely affect paging. The optimal value is dependent on specific system characteristics. You might need to try several values before you determine the ideal buffer size for your system. To change the system wide value, perform the following steps:

**Procedure**
Tune the TCP/IP buffer sizes.

1. Change the TCP/IP configuration.

    a. Run the Change TCP/IP Attribute, **CHGTCPA** command.
    b. View and change the buffer sizes by pressing **F4** on the Change TCP/IP Attributes window. The buffer sizes are displayed as the TCP receive and send buffer sizes. Type new values and save your changes.

2. Recycle TCP/IP, and then monitor CPU and paging rates to determine if they are within recommended system guidelines.

Repeat this process until you determine the ideal buffer size.

### OS level settings

Ideal OS level tuning settings for web sphere can be found at the below link

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/tprf_tuneopsys.html

### JVM settings

Ideal JVM settings for web sphere can be found at the below link

http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/tprf_tunejvm_v61.html

# SUGGESTING CODE MODIFICATIONS

Overview – check the T24 coding standards document and include the performance changes
This section gives some ideas about what do DO and what NOT to DO while writing Online or COB routines for T24.

DO

### Bulk simple transactions in .SELECT routine

The performance of a COB job can be considerably increased by bulking simple transactions. This can be enabled by passing no of records to be bulked in LIST.PARAMETERS<6> before calling BATCH.BUILD.LIST. The no of records specified would be processed in a single transaction boundary, thus improving performance. Care should be taken while bulking transactions. Only record routines processing simple IO (Ex: One write, one delete etc) transactions needs to be bulked else bulking will slow down the performance.

Bulking should not be done if there is lock dependency across multiple sessions for a same record in file, this will end up in deadly embrace situation. Bulking should also not to be done when the record routine does plenty of IO's, this will degrade performance. The criteria which needs to be considered while doing bulking is

Number of IO's per record in JOB.LIST or Size of the Transaction (example if we do 10 UPDATES/INSERTS during a particular job and each record has a size of 4KB and we are writing 40KB per record in JOB.LIST and if we bulk 20 then 40x20 = 800KB per record. If the database couldn't able to flush 800KB faster, then bulking needs to be reviewed and reduced. (This will be part of database tuning; flushing of data depends on IO distribution in hardware level, network bandwidth between application and database, and also depends on redo log size in case of oracle as database. In case of jbase, flushing of data depends on file size, network bandwidth, IO distribution etc). One more factor also needs to be considered while doing bulking is the number of locks taken per record in job list, if jrla is running in shared memory locking mechanism then we should size the locking table accordingly to handle the number of locks.

**Note:-** It always better to start with small number and slowly increase the bulking to an acceptable value after a number of COB runs.

Eg - The control list POS.MVMT.HIST in EB.PRIME.FILES job does a single delete in record routine and for this we are updating JOB.LIST for every record which created performance problem in EB.PRIME.FILES job. Hence we did Bulking of transactions (50 for POS.MVMT.HIST and 10 for POS.MVMT.TODAY) as follows.

D003 – Before Bulking
FILE.TIDY.UP
EB.PRIME.FILES

START TIME: 16:26:36
END.TIME: 17:25:32
ELAPSED TIME: 0:58:56

D003 – After Bulking

FILE.TIDY.UP
EB.PRIME.FILES
START TIME: 8:38:33
END.TIME: 8:48:25
ELAPSED.TIME: 0:09:52

The following changes were done accordingly in EB.PRIME.FILES.SELECT.

```
LIST.PARAMETERS = ''
BEGIN CASE
CASE CONTROL.LIST<1,1> = 'POS.MVMT.HIST'
   LIST.PARAMETERS<2> = 'F.POS.MVMT.HIST'
   LIST.PARAMETERS<7> = 'FILTER'
   LIST.PARAMETERS<6> = 50
CASE CONTROL.LIST<1,1> = 'POS.LWORK.DAY'
   LIST.PARAMETERS<2> = 'F.POS.MVMT.TODAY'
   LIST.PARAMETERS<6> = 10
CASE CONTROL.LIST<1,1> = 'CLOSING.OPEN.BAL'
   LIST.PARAMETERS<2> = 'F.POSITION.LWORK.DAY'
END CASE
```

### Index frequently used Fields

The frequently queried fields should be indexed properly to improve the performance.
The following points should be considered for indexing

- Maximum 4 indexes can be created for a table. (2- 4 MAX)
- For heavily updated but lightly queried tables fewer indexes the better (2) otherwise update will slow down so transaction performance will be reduced.
- For lightly updated tables but heavily queried tables, i.e. Customers, Category,? etc then additional indexes also make sense....(4 max)

It's also recommended to create indexes where ever necessary for the select time consuming COB jobs.

Minimum of two indexes needs to be created for R8

1. NEXT.EXP.DATE field in ACCOUNT table (AC.FWD.EXPOSURE job selects using NEXT.EXP.DATE)
2. NEXT.STMT.DATE field in ACCOUNT.STATEMENT table (PRINT.ACCT.STMT job select using NEXT.STMT.DATE)

### .FILTER routine (Filter id alone)

Before updating JOB.LIST with records needs to be processed, it's always better to use a .FILTER routine to filter the records needs to be processed. And filtration needs to be done based on ID ONLY.

E.g.: LMM.STATIC.UPDATE.FILTER
The best place to use filter routine is when we want to select a huge file and if the conditional select takes more time (i.e. When primary key index is not properly parsed)

#### When to use and not use .FILTER routine

Ex: SELECT FBNK.ACCOUNT WITH @ID UNLIKE …3A… (In this case primary key index will not be used so better we do the .FILTER routine) but if the select properly uses the index then instead of filter it is better to leave database to give the results

Ex: SELECT FBNK.ACCOUNT @WITH @ID LIKE USD… (In this case primary key index will be used and select will be much faster)

### CACHING DAS queries

The DAS infrastructure allows the caching of results by setting:

ADD.TO.CACHE = 1

Once the query results have been returned, they are added to the cache mechanism and further invocation of the named select will always use the cached results. The cache must ONLY be used where the result of the query is static at run time, e.g. CATEGORY or COMPANY or TRANSACTION etc. Caching must NOT be used where variable/dynamic data is used in the selection.

### DO – Key only File – Keep info on Id itself

Large volume concat files, it's better to keep information on the Id (primary key) itself (as much as possible). Since a default index would be created already for ids in the database, SELECT would be faster and also no need of reading each and every record. For E.g.: The update of the single field of the file RE.CONSOL.WORK (ADD, REMOVE) can be done in ID of the work file itself. Since Id is being indexed in oracle, select would be faster

### DO – Lock and write records in multisession env (Avoid Unique Constraint in Oracle)

The same file is being updated in multiple sessions without lock which results in unique constraint error in Oracle DB. So care must be taken while writing multi threaded COB jobs

as its runs in multiple sessions before doing any WRITE/DELETE it's recommended to do an F.READU but if the id is unique per session then F.READU can be avoided.

3.2.2.54845 - 372926 - (F.DX.TRADE.WRITE,29) - Fri Jan 11 20:13:48 - FBNK.DX.TRADE.OUT.OF.SYNC - FBNK_DX_T000 - ** Error ** writeRecordAsBLOB: Unable to write key DXTRA0712100007 record ####
3.2.2.54845 - 372926 - (F.DX.TRADE.WRITE,29) - Fri Jan 11 20:13:48 - ** Oracle Error **: ORA-00001: unique constraint (XMLUSER.FBNK_DX_T000_PK) violated
3.2.2.54845 - 164344 - (F.DX.TRADE.WRITE,29) - Fri Jan 11 20:13:49 - FBNK.DX.TRADE.OUT.OF.SYNC - FBNK_DX_T000 - ** Error ** writeRecordAsBLOB: Unable to write key DXTRA0711000004 record ####
3.2.2.54845 - 164344 - (F.DX.TRADE.WRITE,29) - Fri Jan 11 20:13:49 - ** Oracle Error **: ORA-00001: unique constraint (XMLUSER.FBNK_DX_T000_PK) violated

DON'T

### Complex SELECT

Don't use complex queries which will considerably reduce the performance of the SELECT routine and also increase the overall COB time.
E.g.:

```
CASE MY.CMD = dasEntitlementValueBeforeDate ; * Used for archiving
  MY.FIELDS     = 'VALUE.DATE'
  MY.OPERANDS   = 'LE'
  MY.DATA       = THE.ARGS<1>
  MY.JOINS      = 'AND'

  MY.FIELDS<2>  = '(RECORD.STATUS'
  MY.OPERANDS<2> = 'EQ'
  MY.DATA<2>    = 'INAU'

  MY.JOINS<2>   = 'OR'

  MY.FIELDS<3>  = 'RECORD.STATUS'
  MY.OPERANDS<3> = 'EQ'
  MY.DATA<3>    = 'RNAU'

  MY.JOINS<3>   = 'OR'

  MY.FIELDS<4>  = 'RECORD.STATUS' ; * BG_100013322 S
  MY.OPERANDS<4> = 'EQ'
  MY.DATA<4>    = 'IHLD)'       ; * BG_100013322 E
```

### Select in .RECORD routine

SELECT shouldn't be done in RECORD routine which would create performance problems. If there is a select statement in record routine then during this job multiple tSA's will do the select and it needs more processing power. SELECT should be done only in .SELECT routine.

### Select on I-Descriptors

SELECT based on I-Descriptor field could cause a severe performance problem in Oracle. The driver will select all records and then filter would be done based on the code specified in I-Descriptor which would considerably reduce the performance. E.g.: The DAS select 'dasLmmSchedulesWithFullSchDateCondition' based on I-descriptor on ID (FULL.SCH.DATE) was creating performance problems in oracle - then modified to select directly from ID itself.

File DICT F.LMM.SCHEDULES , Record 'FULL.SCH.DATE' Insert
20:41:29
Command->
0001 I
0002 @ID[13,7]
0003
0004 FULL.SCH.DATE
0005 8R
0006 S

### Select on huge files

Do NOT select files that are known to be likely to grow large in installations, use concat files or other alternative methods of finding the records required. When developing new applications / processes. If a process will be dependent upon selection of a file that will grow large, build an alternative access method, e.g. a concat file, index, key only file

Example files include:

i. STMT.ENTRY
ii. CATEG.ENTRY
iii. RE.CONSOL.SPEC.ENTRY
iv. LIMIT.TXNS
v. SC.POS.ASSET

### CONCAT File design

Be very careful in the use of concat files. A poorly designed concat file can result in a performance overhead when updated due to the large record size created and system locks if the key is a poor choice. E.g. the concat file ACCT.ENT.TODAY is a poor file since the key is

likely to be common with the majority of contracts (Internal accounts or nostro accounts), this has the effect of allowing only a single contract to be entered at any one time for a given currency, and is likely to contain massive records (e.g. for local currency). If a concat file is likely to contain more than 1000 ids in it a different structure / process should be used.

The main things which needs to be considered for a concat file

1. The ID of the file should not be lock bound (like account, date, portfolio, currency, SECURITY.MASTER, etc
2. The information which are kept inside the concat file should not grow rapidly (say 100 ids per day, with in 10 days 1000 ids will be updated inside which will slow down the update/insert to the concat file
3. Keep as much as information in the ID itself and make the ID with more variables so that I can't be the same for each transaction.

### Direct Select on Multi value fields (Use a separate concat file)

SELECT based on multi value fields will take much more time than normal field in a XML based database. One of the main disadvantage of multi value field in an RDBMS is indexes can't be created, so queries will be parsed manually at jBASE level.

For E.g.: It took almost 20 minutes to select CUSTOMER.CHARGE. But in the record routine, we are returning if CHARGE.FREQ is null. A direct select with CHARGE.FREQ null in CUSTOMER.CHARGE also took almost 15 minutes. So a concat file needs to be created for this multi valued field with id as CUSTOMER on which the select can been performed which will improve the performance.

### DON'T – Use ID length more than 128 bytes (Oracle accepts only 255 bytes).

Oracle accepts only 255 bytes as Id length (For Sql server its 128). So don't use Id length greater than 128 which would result in the crash of DB as follows.
3.2.2.54845 - 331928 - (jutil.copycommand.b,281) - Mon Jan 7 12:06:46 - FBNK.TXN.JOURNAL - FBNK_TXN_JOURNAL_1 - ** Error ** WRITE: Record id length 256 exceeds maximum of 255 bytes. Key = 1*GB0010001*20070827*MG*NET-R!MG.1.TR.USD.25002.1001.AU.5Y..PDO..1000.51000SP.20070919!CR!USD!MG!ACC!20070827!!!!!!20070827!2620!25002*25002.1001.A*R*R!MG.1.TR.USD.25002.1001.AU.5Y..PDO..1000.51000SP.20070919!CR!USD!MG!ACC!20070827!!!!!!20070827!2620!25002

### DON'T - Call Multiple DBR's for same file

Try to avoid DBR to get the value of a field. If there is a situation of using multiple DBR's for a same file, then use F.READ instead.

### DON'T - Use CACHE.READ for dynamic tables like ACCOUNT, CUSTOMER etc

CACHE.READ should be called only for static tables in T24 like Parameter files, PGM.FILE etc. For dynamically changing tables like ACCOUNT, CUSTOMER, SEC.ACC.MASTER etc, F.READ should be used instead.

### DON'T – Call CACHE.READ and F.READ without ID

Care should be taken while calling CACHE.READ and F.READ. Lot of routines are there calling CACHE.READ and F.READ without ID.

### Don't lock records if we're updating the id with unique session number

For large tables, it is always better to use session number in Id. By using session number in ID, the unnecessary locking of files can be avoided.

For example, EB.UPD.CONSOL.UPDATE.WORK, we're locking and updating the workfile. And if there is a lock, we're updating the work file with session number. Instead, If we use session number directly, the locking can be avoided (can use F.READ instead of F.READU) which in turn improves the performance

## CASE STUDY

What follows is a case study that elegantly demonstrates that the analytical principles outlined in this document. The case under examination occurred on the run up to a UAT cycle of a mid-tier retail bank.

Background
The T24 system under test consisted of some 14 Million accounts, 13 million customers with several hundred thousand teller and half a million clearing entries per day. Online performance was adequate but the Close of Business was required to complete in six hours on a normal day, and allowed to extend to eight hours for a capitalisation run.
The initial Close of Business took 18 hours to complete, with the accrual throughput running at approximately 62,300 accounts per minute.
Initial investigation immediately revealed network bottlenecks, display the following symptoms:

- Addition of extra worker threads did not increase throughput
- Addition of extra work threads did not increase application or database server CPU usage (which was barely breaking 20% on each machine)

- Memory usage was well within bounds
- The top wait events in Oracle were network wait events
- The network usage was peaking at, but never exceeding the performant capacity of a single gigabit card (see “Dummies Guide to Networks” for details)
- The operating system monitor showed alerts of network bottleneck
- Network monitoring tools showed packet loss and dropped connections

Steps

### Step 1 – Network Tuning

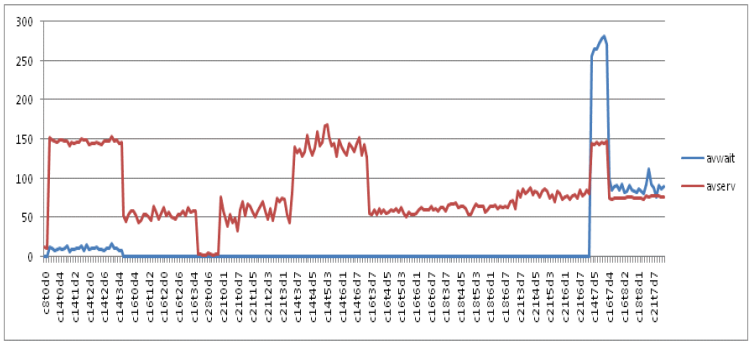
Given the above evidence, extra network cards were added increasing the bandwidth from a single gigabit card to a total of four gigabit cards. Initially this did not improve performance, until the network settings of both the application server and database server were tuned. Once accomplished, an accrual close of business was run.
The total elapsed time was eight hours.

### Step 2 – Task Reduction and Database Tuning

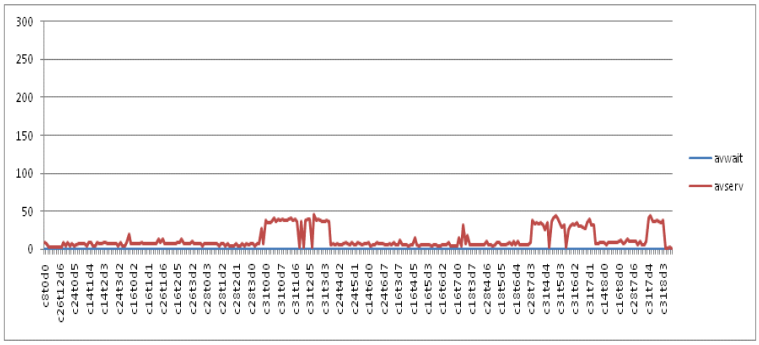
At this point the network bottleneck was removed, and another factor came into play. Here we started to witness that IO was an issue. Previously, disk response times were acceptable (service times within the 8-10 ms range with 0 wait time). However, once T24 was able to push Oracle thanks to the removal of the network bottleneck
At this point the total elapsed time was five hours.

### Step 3 – Reconfigure Storage

The disk array was re-configured to the recommended settings of a one megabyte stripe across all disks. This dramatically increased the response times of the disk system, as per the below graphs.




**Disk response times with 8Kb stripe, in milliseconds.**




**Disk response times with 1Mb stripe, in milliseconds.**

### Step 4 – Move Oracle Tables to ASSM

Moving highly used tables like ACCOUNT, EB.CONTRACT.BALANCES to ASSM tablespace showed lot of improvement during COB.

## CONCLUSION

Although an extreme example, this case studies clearly shows that the overall performance of T24 is dramatically affected by the configuration of the IT infrastructure around it. By using a methodical approach, and realising that there is only one limiting factor at play at any given time, it was possible to remove the bottlenecks until the required performance was achieved.