Oracle Rdb
Technical Forums

Tracking Rdb Performance Using T4

Keith W. Hare
JCC Consulting, Inc.

Abstract

- VMS V7.3-2 (and later) includes a set of tools for capturing VMS performance information in Comma Separated Variable length (CSV) T4 format files that can be viewed using any tool or utility that reads CSV data.
- It is also possible to process RMU/SHOW STATISTICS binary output files to produce T4 format CSV files.
- The combination of both VMS and Rdb T4 data provides a low-overhead mechanism for capturing and reviewing a larger number of performance measurements that can be used to understand current database performance trends and possibly predict and avoid future performance bottlenecks.
- Specifically, this session presents:
  - T4 Tools
  - Capturing VMS and Rdb T4 data
  - Measuring the overhead of Capturing VMS and Rdb T4 data
  - Integrating VMS and Rdb data
  - Identifying performance opportunities using T4 data
  - Predicting future performance bottlenecks

Introduction

The topics covered in this presentation are:

- Why do you Care?
- T4 Tools
- Capturing VMS and Rdb T4 data
- Measuring the overhead of Capturing VMS and Rdb T4 data
- Integrating VMS and Rdb data
- Identifying performance opportunities using T4 data
- Predicting future performance bottlenecks
- Additional Opportunities

Why Do You Care?

The T4 performance tools support:

- Low impact mechanism for gathering VMS and Rdb performance information
- Historical Archive of performance information

T4 Tools

There are a number of T4-related tools:

- What is T4?
- VMS T4 Tools
- TLVis – Timeline Visualization
- RMU/Show Statistics/output=
- Command Procedure to Manage RMU Stats
- RMU to T4

What is T4?

From the VMS T4 Documentation:

> *T4 (Tabular Timeline Tracking Tool) is a cooperating collection of low overhead timeline utilities. T4 systematically captures, consolidates, and creates a composite timeline view and an historical archive of important OpenVMS performance statistics. T4 can help insure that a detailed, day-by-day performance history of each node is available whenever it might be needed. T4 simplifies the job of observing how the performance of a single OpenVMS node varies over time.*

- Free performance data collector
- Low system impact
- Officially unsupported
- Easy to use
- Very useful

VMS T4 Tools

The VMS T4 kit is included with VMS V7.3-2 and later:

```
atlas > di sys$etc:t4*
Directory SYS$COMMON:[SYSHLP.UNSUPPORTED]
T4_V33_KIT.EXE;1         705/735       1-OCT-2003 21:19:45.97
   (RWED,RWED,RE,RE)
T4_V33_KIT.TXT;1          65/70       14-MAY-2003 06:38:38.43
   (RWED,RWED,RE,RE)
Total of 2 files, 770/805 blocks.
atlas >
```

For VMS V7.2-2, V7.3, and V7.3-1, a kit can be downloaded from the HP Web Site:

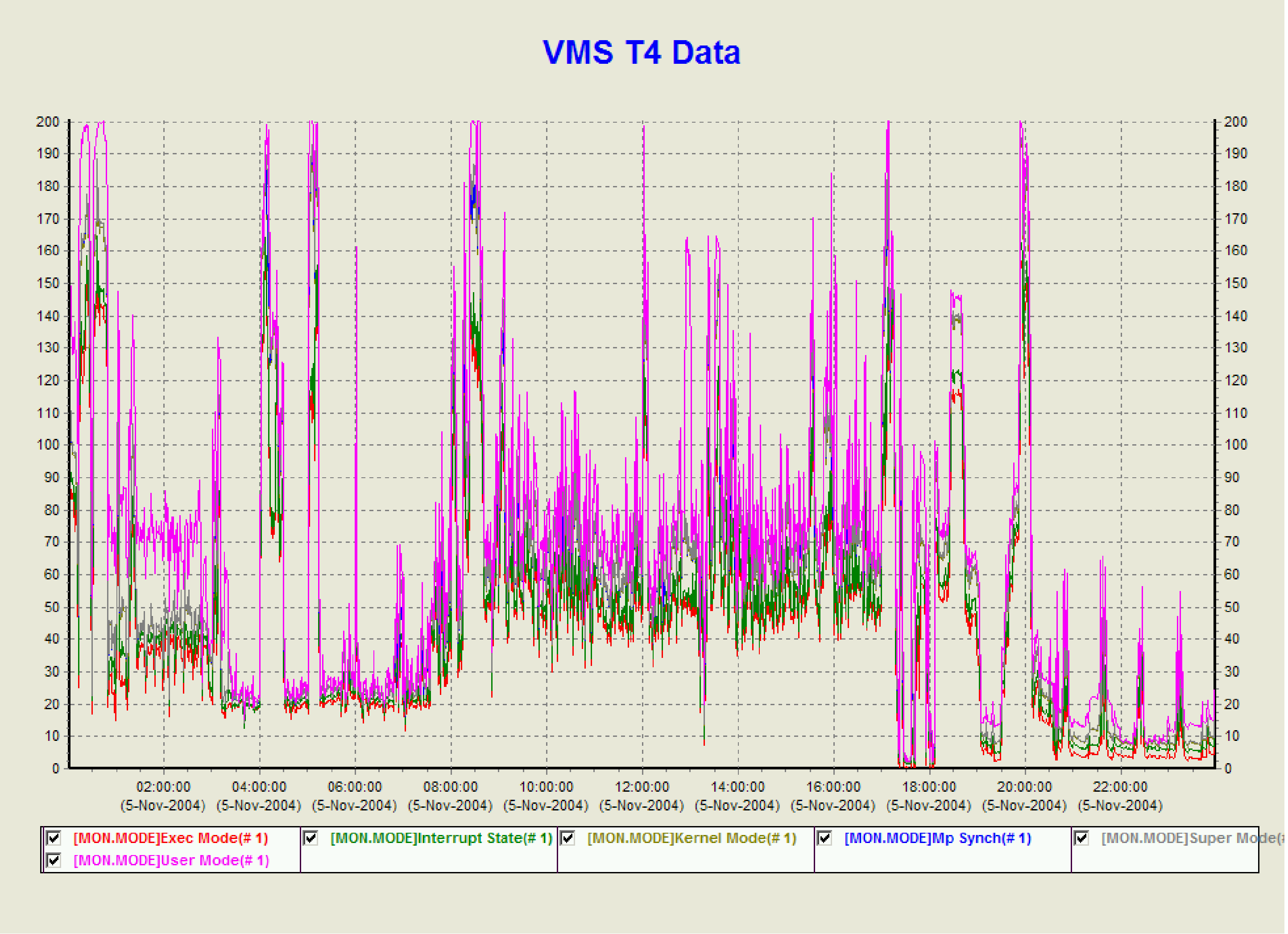
http://h71000.www7.hp.com/OpenVMS/products/t4/index.html

TLVis – Timeline Visualization

T4 Format data is in a Comma Separated Variable length (CSV) file.

- Can be read by any tool that can read CSV files, Excel for example.
- TLVis is a PC based TimeLine Visualization Tool
    - Understands T4 format data
    - Fast
    - Unsupported tool from VMS engineering
    - Free
    - Available on the Rdb Technical Forum CD

TLVis Example – CPU Modes



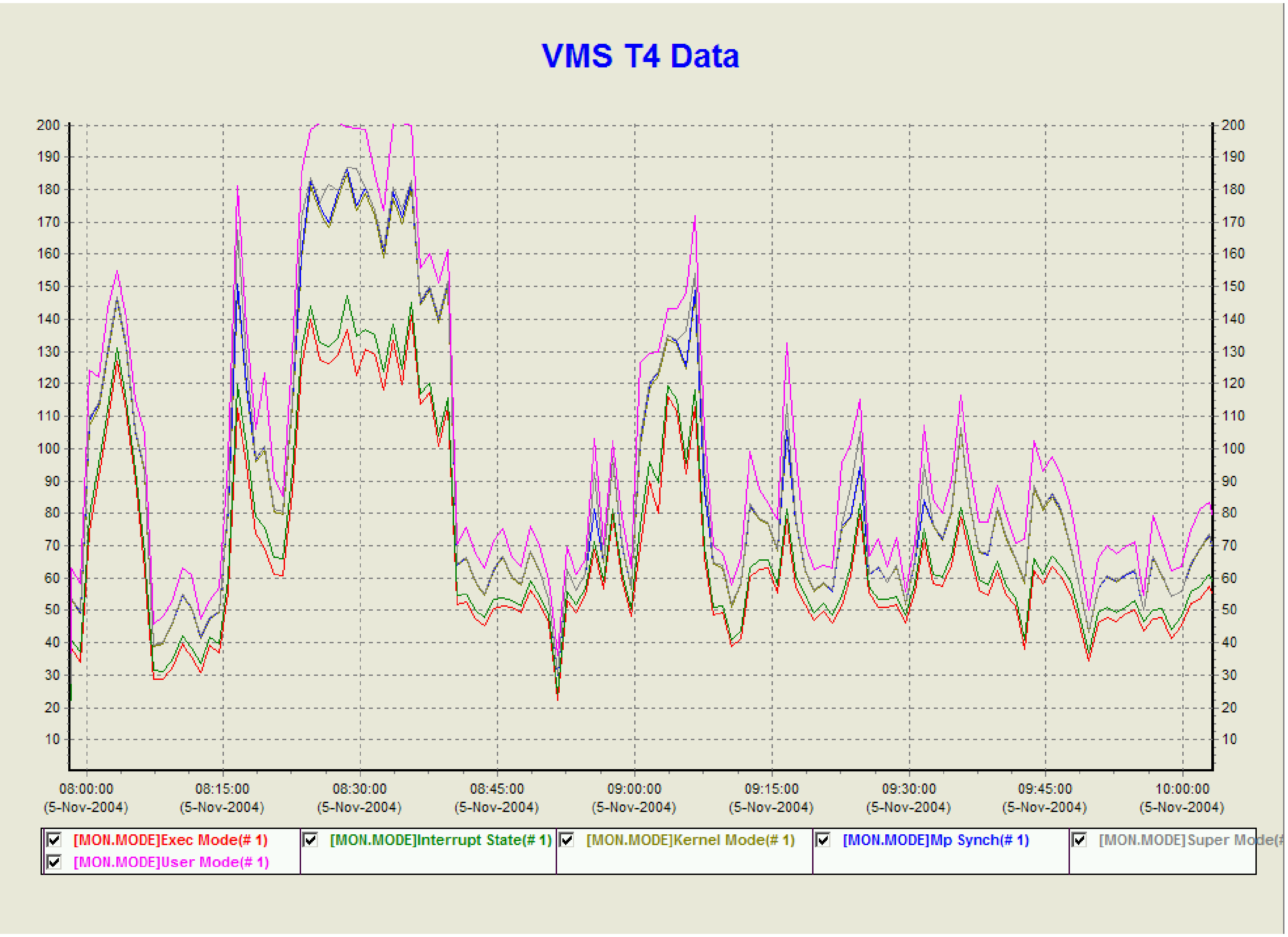
CPU Modes – Comments

The previous graph shows the amount of CPU time spent various modes on a 2-cpu ES40.

- Exec Mode – Rdb, RMS, and DBMS
- Interrupt State
- Kernel Mode
- MP Synchronization
- Supervisor Mode
- User Mode

The peaks at 200% indicate points where all of the CPU is being used.

TLVis Example – Zoom

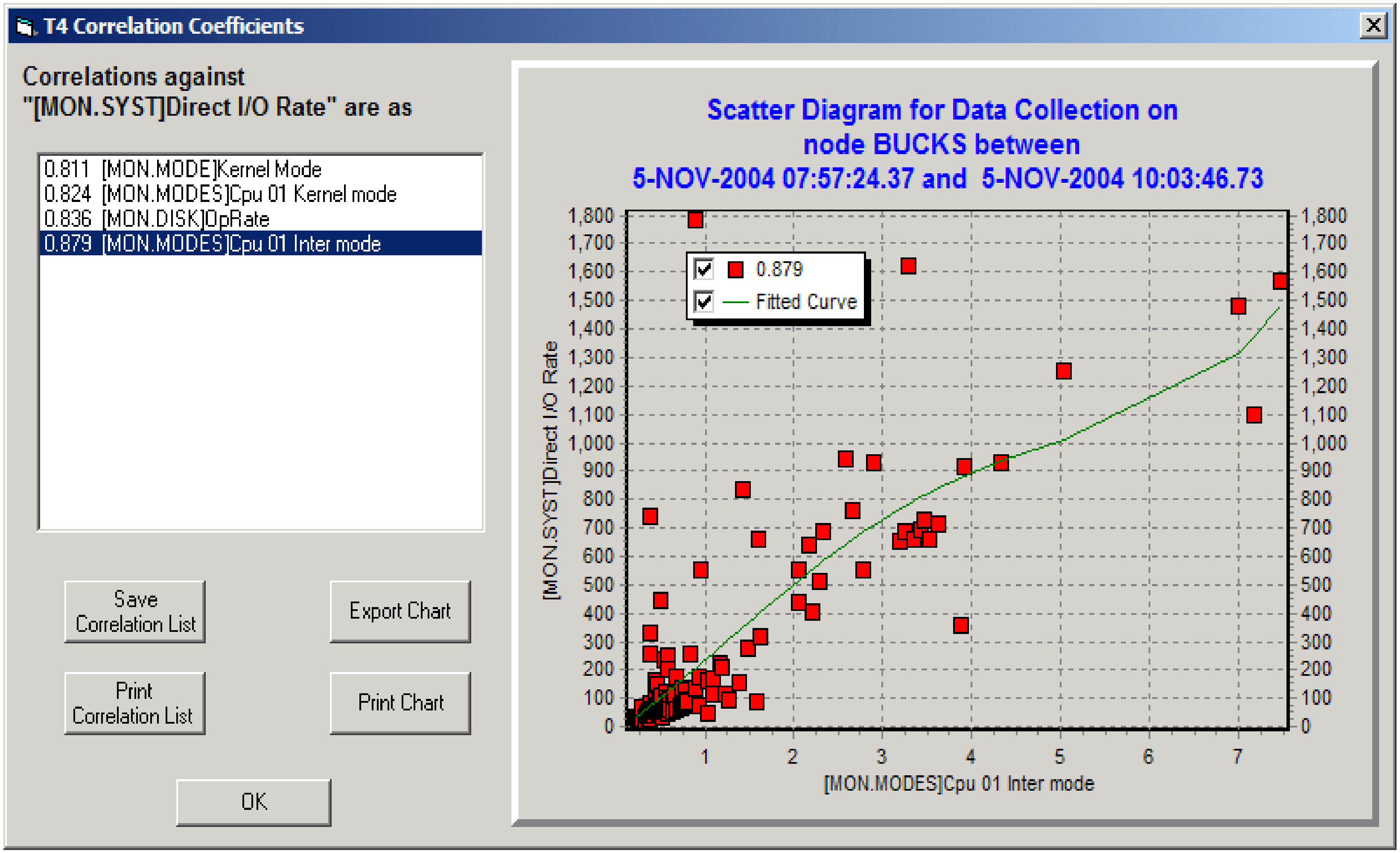


Zoom Comments

This graph show the CPU modes between 08:00 and 10:00. This is the same data as the previous CPU modes graph.

- Zoom level crosses statistics
- Can unzoom

TLVis Correlation Coefficients



T4 Correlation Coefficients

Correlations against
"[MON.SYST]Direct I/O Rate" are as

```
0.811  [MON.MODE]Kernel Mode
0.824  [MON.MODES]Cpu 01 Kernel mode
0.836  [MON.DISK]OpRate
0.879  [MON.MODES]Cpu 01 Inter mode
```

Scatter Diagram for Data Collection on
node BUCKS between
5-NOV-2004 07:57:24.37 and 5-NOV-2004 10:03:46.73

Correlation Coefficients – Comments

TLVis will identify statistics that correlate with the current statistics. In this example, [MON.SYSTEM]Direct I/O Rate has:

- An .879 correlation with CPU 1 interrupt state
  - Suggests that I/Os are handled by CPU 1
- An .836 correlation with disk operations
- An .824 correlation with CPU1 Kernel mode
  - Suggests the I/Os happen in kernel mode

RMU/Show Statistics/output=

RMU/Show Statistics can be run in batch go collect data in a binary output file:

```
$       rmu     /show statistics  -
                /nointeractive    -
                /nological_area   -
                /nobroadcast      -
                /time=<monitor frequency> -
                /output=<output filename> -
                /until=<stop time>        -
                <database_name>
```

Command Procedure to Manage RMU Stats

The Rdb Technical Forum CD includes the following command procedure:

CAPTURE_RMU_SHOW_STAT_BINARY_OUTPUT.COM

- Parameters:
  - P1 – Database Name, no default
  - p2 – Frequency of monitoring, default 60 second
  - p3 – prefix for binary output files
  - p4 – end time for monitoring. If CONT, reopens output file at midnight
- Example:

```
$ submit/noprint/log=[] -
   USER_ROOT:[JCC.DBA_TOOLS]CAPTURE_RMU_SHOW_STAT_BINARY_OUTPUT -
 /param=(scp_production_db, 60, -
         USER_ROOT:[JCC.RMU_STATS]prod_db, cont)
```

RMU to T4

RMU/Show Statistics output files can be converted to T4 format CSV files using the program PerfT4:

```
atlas > cc perft4
atlas > link perft4
atlas > perft4 == "$ JCC_ROOT:[KEITH.T4]perft4"
atlas > perft4 SCP_PROD_DB_RMU_STATS_2004_11_22.dat
   SCP_PROD_DB_RMU_STATS_2004_11_22.csv
database DISK$DATABASE:[PRODUCTION_DB]SCP_DB_ROOT.RDB;1
                            on node BUCKS opened 07-Nov-2004 22:37:06
max users=250, max nodes=1, areas=1436, max logical areas=1838
Wrote 297 records (from 22-Nov-2004 12:20:55 to 22-Nov-2004 17:18:10)
atlas >
```

This version of PerfT4.C was downloaded from Metalink Note 282894.1

Capturing VMS T4 data

```
atlas > @t4$sys:T4$CONFIG

Copyright 2000-2003 Hewlett-Packard Development Company, L.P.

Executing T4$CONFIG.COM on node ATLAS - Date/Time is now 24-NOV-2004 14:57:50.66
Collection Start Time [25-NOV-2004 08:00:00.00]         : 25-Nov-2004 00:00:00
Collection End Time   [25-NOV-2004 20:00:00.00]         : 25-Nov-2004 23:59:59.99
Batch queue name                      : sys$batch
Network Interface Device (? for list, type RETURN to finish) : ?
Names of Network Interface devices on this system are :-

EWC0:
EWA0:
EWB0:
EWD0:

Network Interface Device (? for list, type RETURN to finish) : EWC0
Network Interface Device (? for list, type RETURN to finish) : EWA0
Network Interface Device (? for list, type RETURN to finish) : EWB0
Network Interface Device (? for list, type RETURN to finish) : EWD0
Network Interface Device (? for list, type RETURN to finish) :
Sampling Interval (seconds) [60]      :
Setting SAMPLING Interval to default of 60
Destination Directory [T4$SYS]        :
Automatically manage T4 data storage        [N] : y
Number of days to retain raw data           [7] :
Number of days to retain intermediate files [3] :
Number of days to retain reduced files   [9999] :
Re-Submit data collection job daily         [N] : y
Email address                                   :
Job T4$COLLECT_V33 (queue ATLAS_BATCH, entry 932) holding until 24-NOV-2004 23:58
atlas >
```

Overhead of Capturing VMS T4 data

Gathering the VMS data once every 60 seconds requires very little overhead:

```
bucks ~core~03.01> show time
  24-NOV-2004 13:43:06
bucks ~core~03.01> pipe show sys | search sys$pipe t4
00000EC4 T4$V33_240000   HIB      6     2343   0 00:00:00.53      1338    230  B
bucks ~core~03.01> pipe show sys/sub | search sys$pipe 00000EC4
00000EC9 00000EC4_MON    LEF     15    19737   0 00:00:05.31       277    390  S
00000ECA 00000EC4_XFC    HIB     15     2572   0 00:00:00.40       462    213  S
00000ECB 00000EC4_Lck7   HIB     15     2573   0 00:00:00.39       461    206  S
00000ECC 00000EC4_TCP    HIB     15     5093   0 00:00:00.41       541    304  S
00000ECD 00000EC4_EWB0:  HIB     15     3460   0 00:00:00.26       551    210  S
00000ECE 00000EC4_EWA0:  HIB     15     3457   0 00:00:00.34       551    210  S
bucks ~core~03.01>
```

In 13.5 hours, the T4 processes have used less than 10 seconds of CPU time.

Overhead of Capturing Rdb T4 data

Gathering RMU statistics once every 60 seconds requires slightly more overhead:

```
bucks ~core~03.01> show proc/account/id=0000044E

24-NOV-2004 13:47:03.72   User: SCP_PROD         Process ID:   0000044E
                          Node: BUCKS            Process name: "Save Rdb Stats""

Accounting information:
 Buffered I/O count:       406  Peak working set size:      47520
 Direct I/O count:         834  Peak virtual size:         308128
 Page faults:            91989  Mounted volumes:                0
 Images activated:          10
 Elapsed CPU time:          0 00:01:13.93
 Connect time:              1 20:01:25.07

Soft CPU Affinity: off
bucks ~core~03.01>
```

This process has used 1 minute, 14 seconds of CPU time in 44 hours.

Integrating VMS and Rdb data

T4 tools include a utility to merge T4 format CSV files

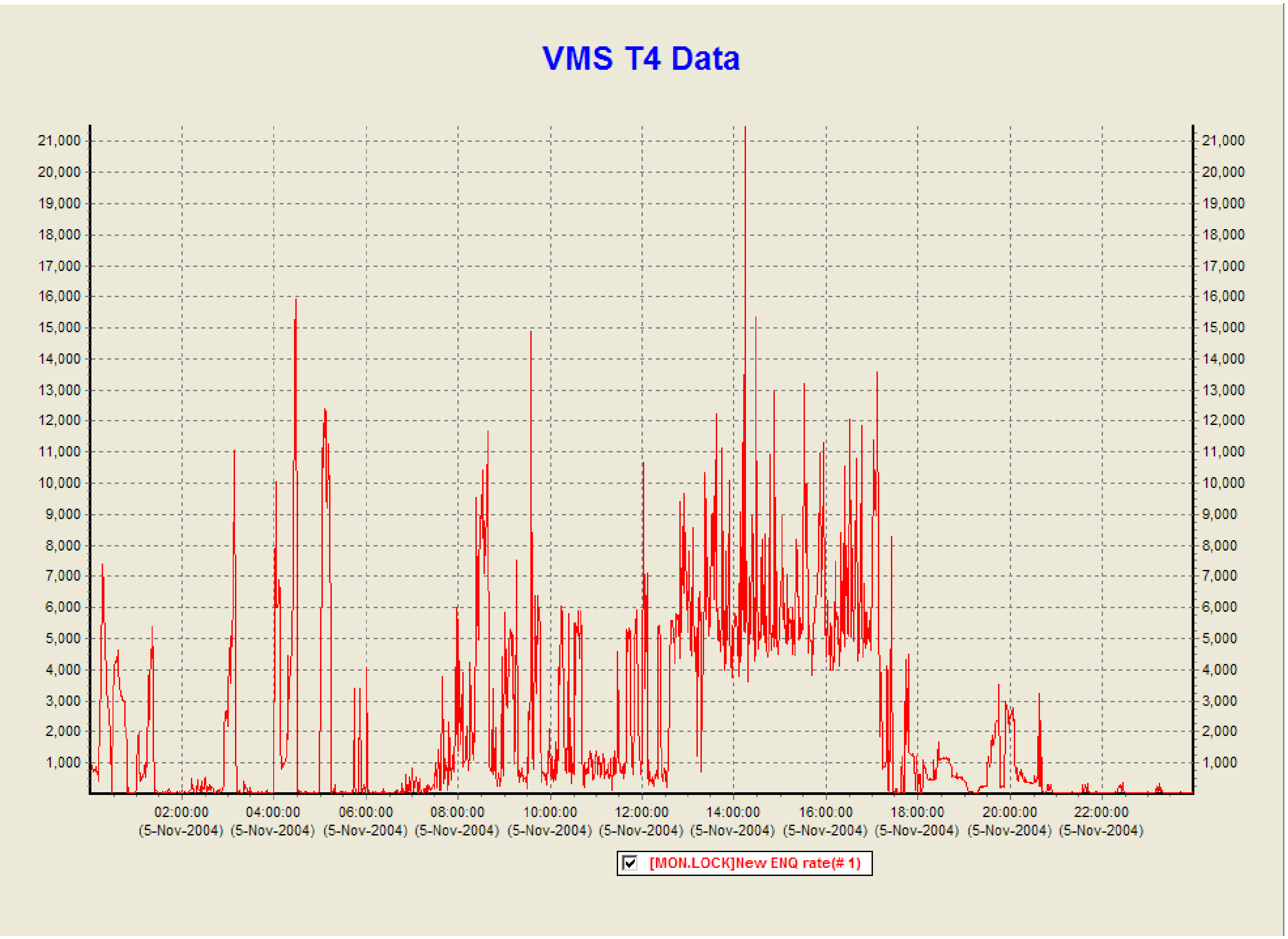
- Only makes sense if files:
  - Start at the same time
  - Use the same interval
- So far, I have not found this to be useful
  - Often working with Rdb and VMS T4 data that starts at different times
  - Too many fields
- Use Multiple TLVis windows, zoomed to cover the same time periods

Identifying Performance Opportunities Using T4 Data

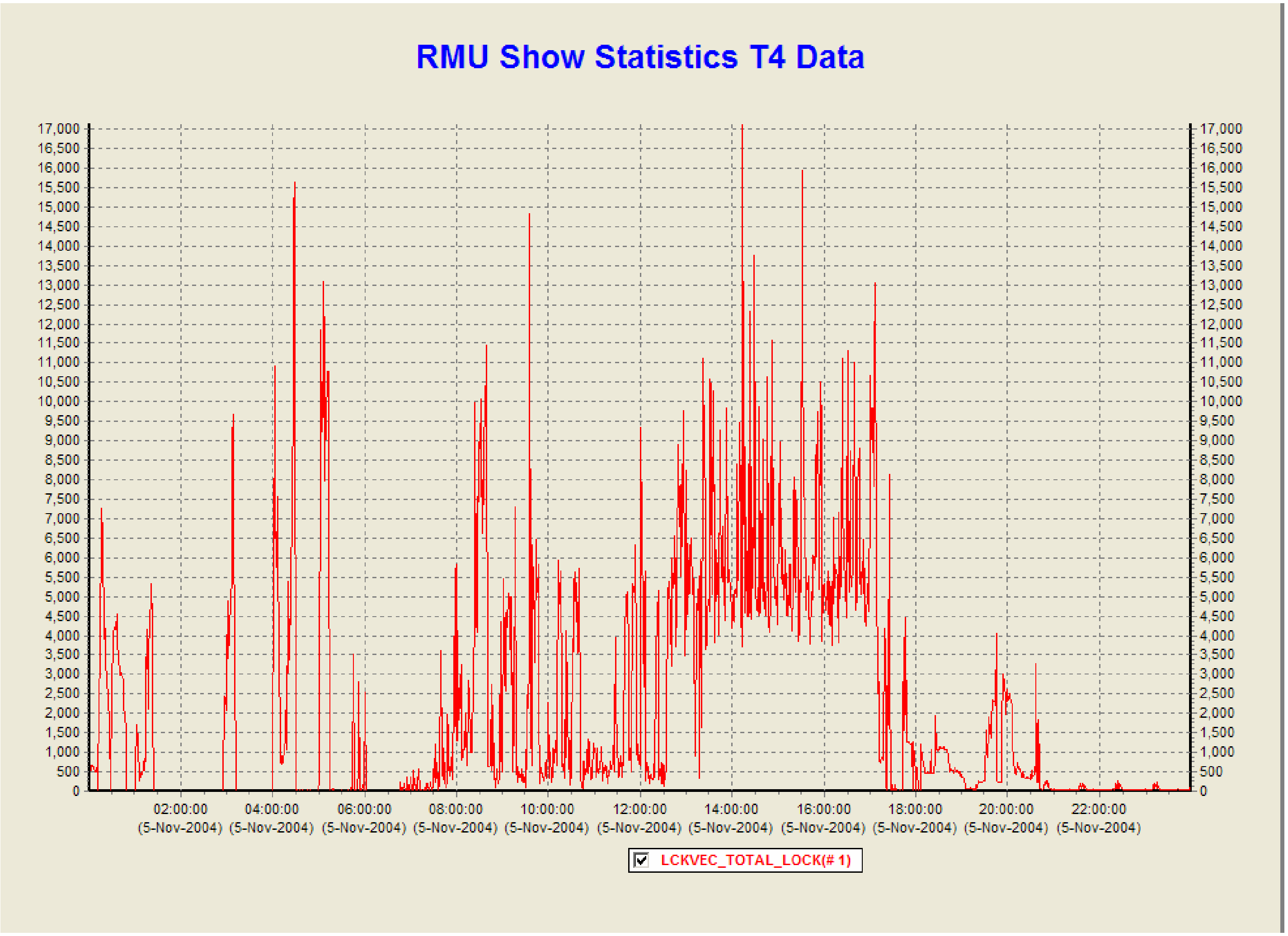
The following slides show examples of how the T4 VMS and RMU data can be used to identify performance opportunities.

- VMS and Rdb Locks
- VMS and Rdb I/O
- Fragmented Records
- Frequent Attaches
- Row Cache Hit Rates

VMS Locks


VMS T4 Data
21,000
20,000
19,000
18,000
17,000
16,000
15,000
14,000
13,000
12,000
11,000
10,000
9,000
8,000
7,000
6,000
5,000
4,000
3,000
2,000
1,000
02:00:00 (5-Nov-2004)
04:00:00 (5-Nov-2004)
06:00:00 (5-Nov-2004)
08:00:00 (5-Nov-2004)
10:00:00 (5-Nov-2004)
12:00:00 (5-Nov-2004)
14:00:00 (5-Nov-2004)
16:00:00 (5-Nov-2004)
18:00:00 (5-Nov-2004)
20:00:00 (5-Nov-2004)
22:00:00 (5-Nov-2004)
☑ [MON.LOCK]New ENQ rate(# 1)

Rdb Locks

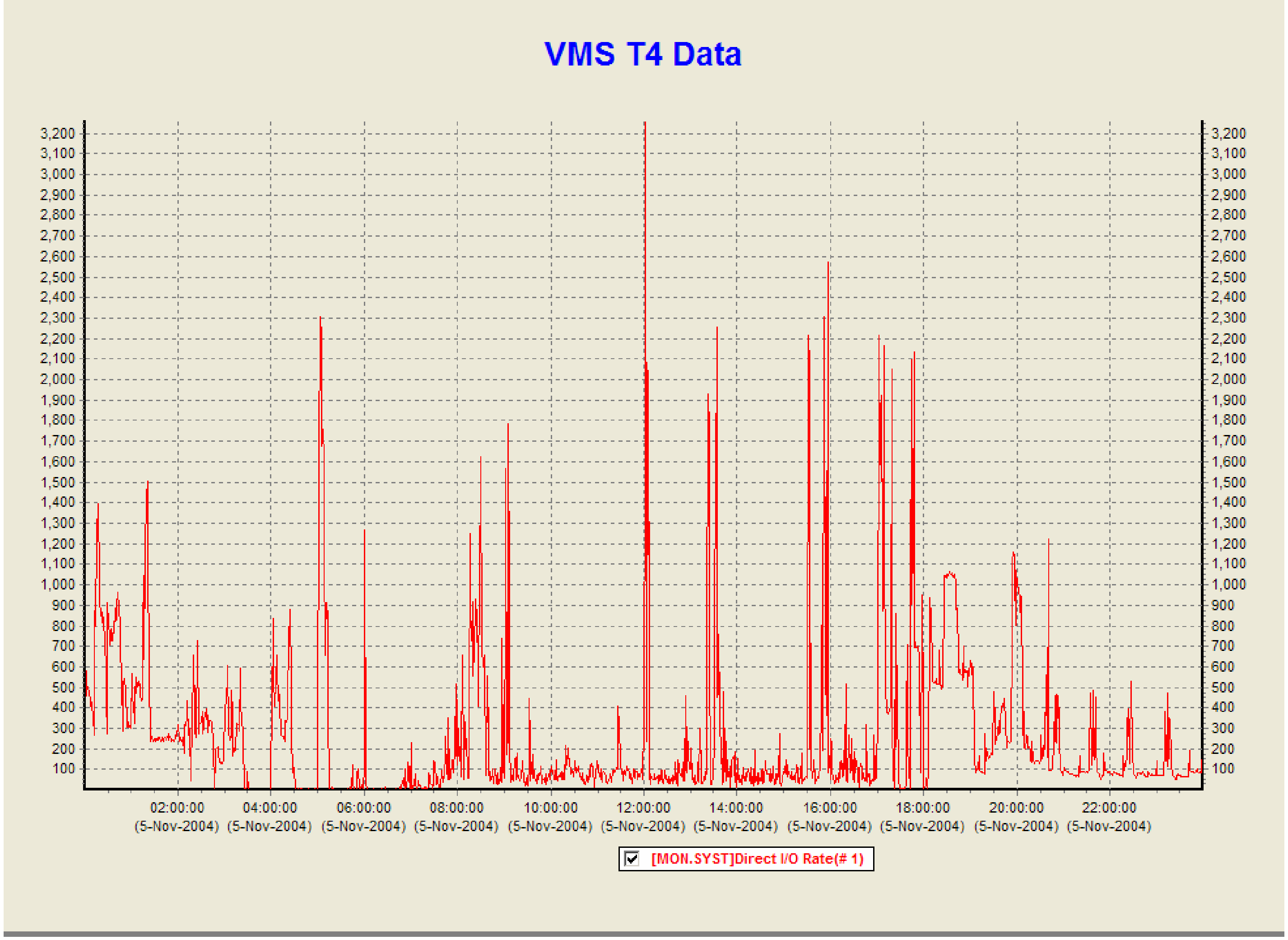

RMU Show Statistics T4 Data
17,000
16,500
16,000
15,500
15,000
14,500
14,000
13,500
13,000
12,500
12,000
11,500
11,000
10,500
10,000
9,500
9,000
8,500
8,000
7,500
7,000
6,500
6,000
5,500
5,000
4,500
4,000
3,500
3,000
2,500
2,000
1,500
1,000
500
0
02:00:00 (5-Nov-2004)
04:00:00 (5-Nov-2004)
06:00:00 (5-Nov-2004)
08:00:00 (5-Nov-2004)
10:00:00 (5-Nov-2004)
12:00:00 (5-Nov-2004)
14:00:00 (5-Nov-2004)
16:00:00 (5-Nov-2004)
18:00:00 (5-Nov-2004)
20:00:00 (5-Nov-2004)
22:00:00 (5-Nov-2004)
☑ LCKVEC_TOTAL_LOCK(# 1)

Comments on Locks

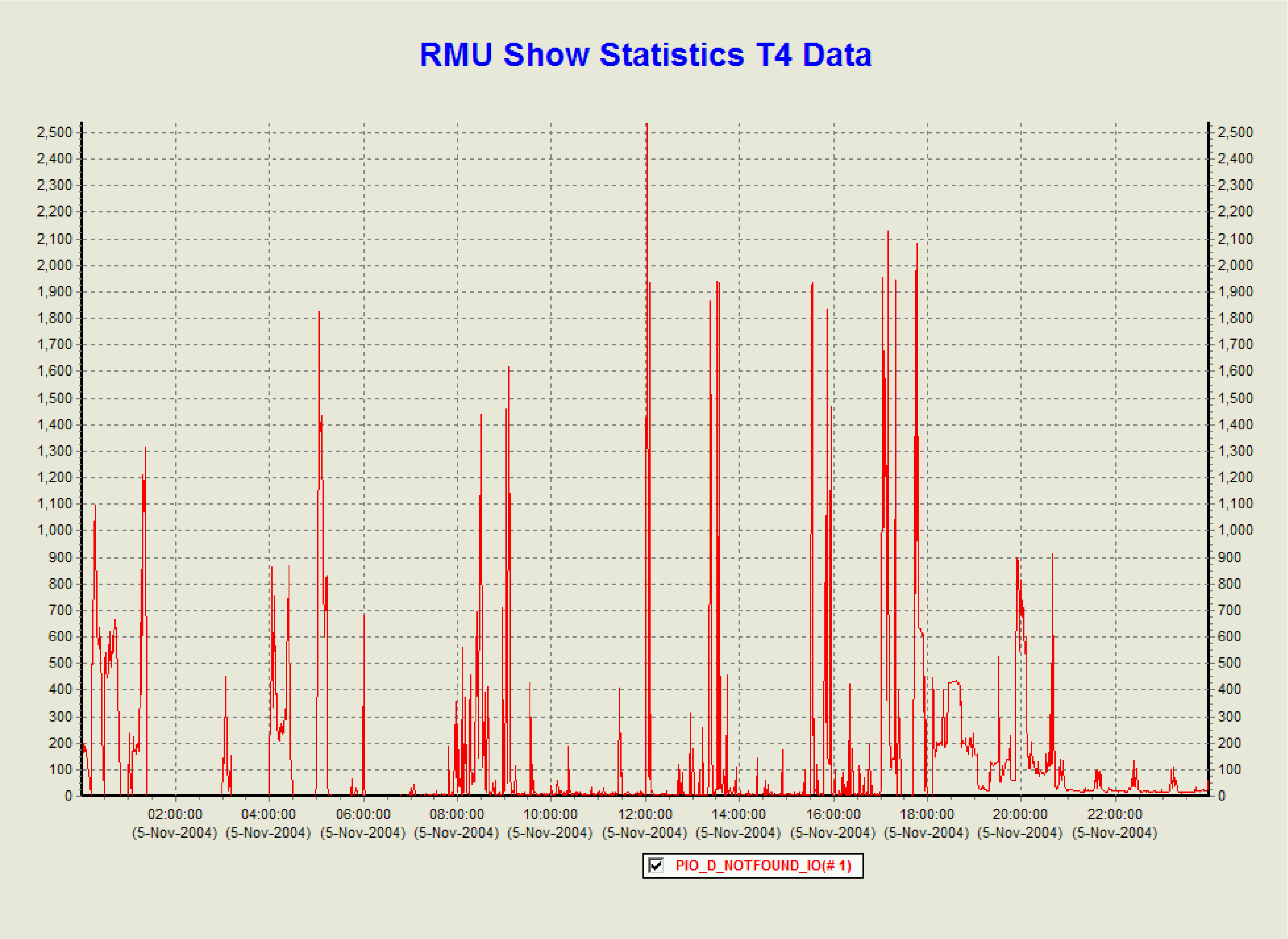
The graphs of VMS Lock operations and Rdb lock operations are very similar.

- Rdb is the largest user of lock operations
- Heavy usages between 12:30 and 17:30
- Sharp peaks suggest that locking operations are not a bottleneck.

VMS Direct I/Os



Rdb I/O

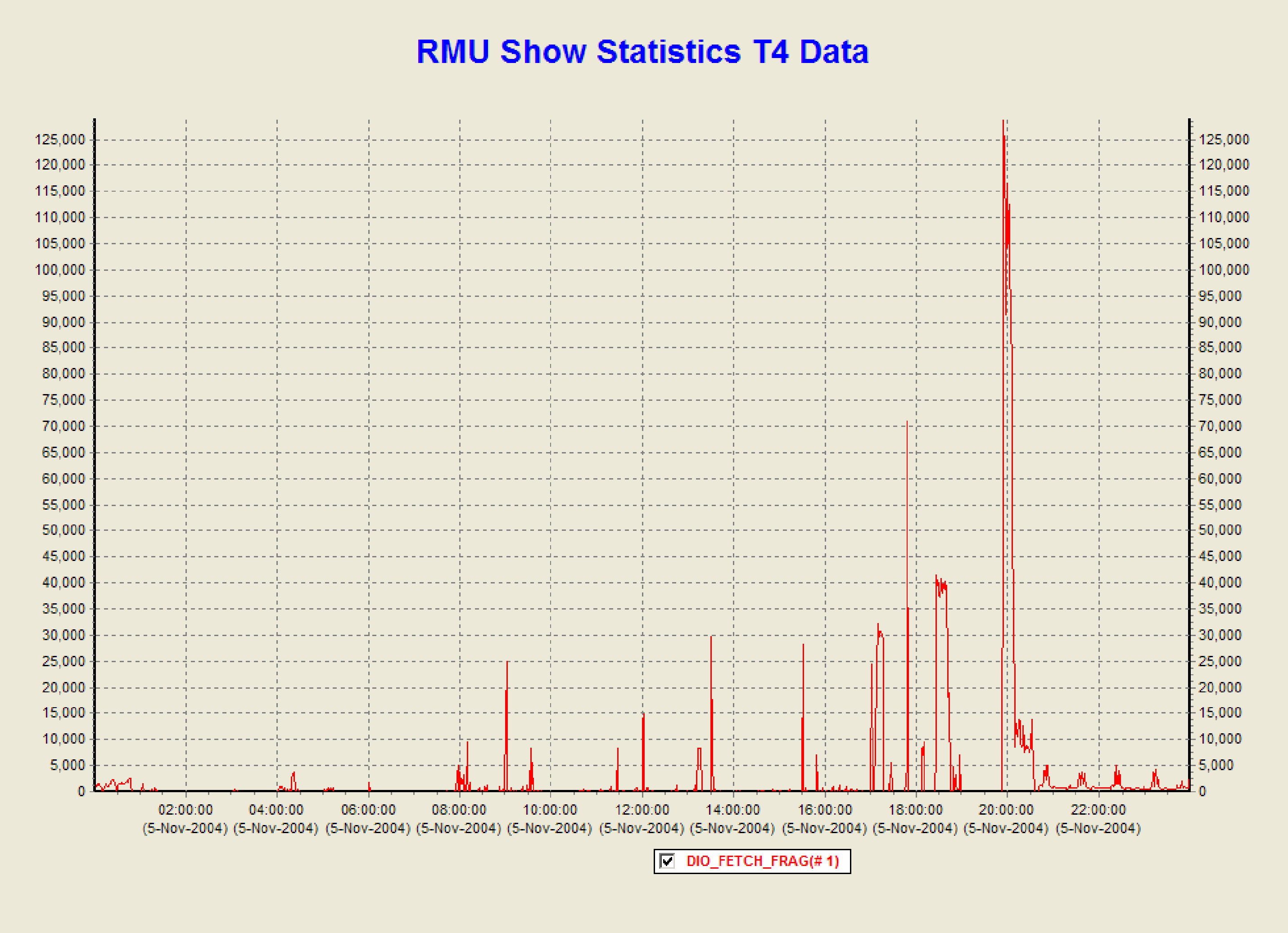

RMU Show Statistics T4 Data
2,500
2,400
2,300
2,200
2,100
2,000
1,900
1,800
1,700
1,600
1,500
1,400
1,300
1,200
1,100
1,000
900
800
700
600
500
400
300
200
100
0
02:00:00 (5-Nov-2004)
04:00:00 (5-Nov-2004)
06:00:00 (5-Nov-2004)
08:00:00 (5-Nov-2004)
10:00:00 (5-Nov-2004)
12:00:00 (5-Nov-2004)
14:00:00 (5-Nov-2004)
16:00:00 (5-Nov-2004)
18:00:00 (5-Nov-2004)
20:00:00 (5-Nov-2004)
22:00:00 (5-Nov-2004)
☑ PIO_D_NOTFOUND_IO(# 1)

Comments on I/Os

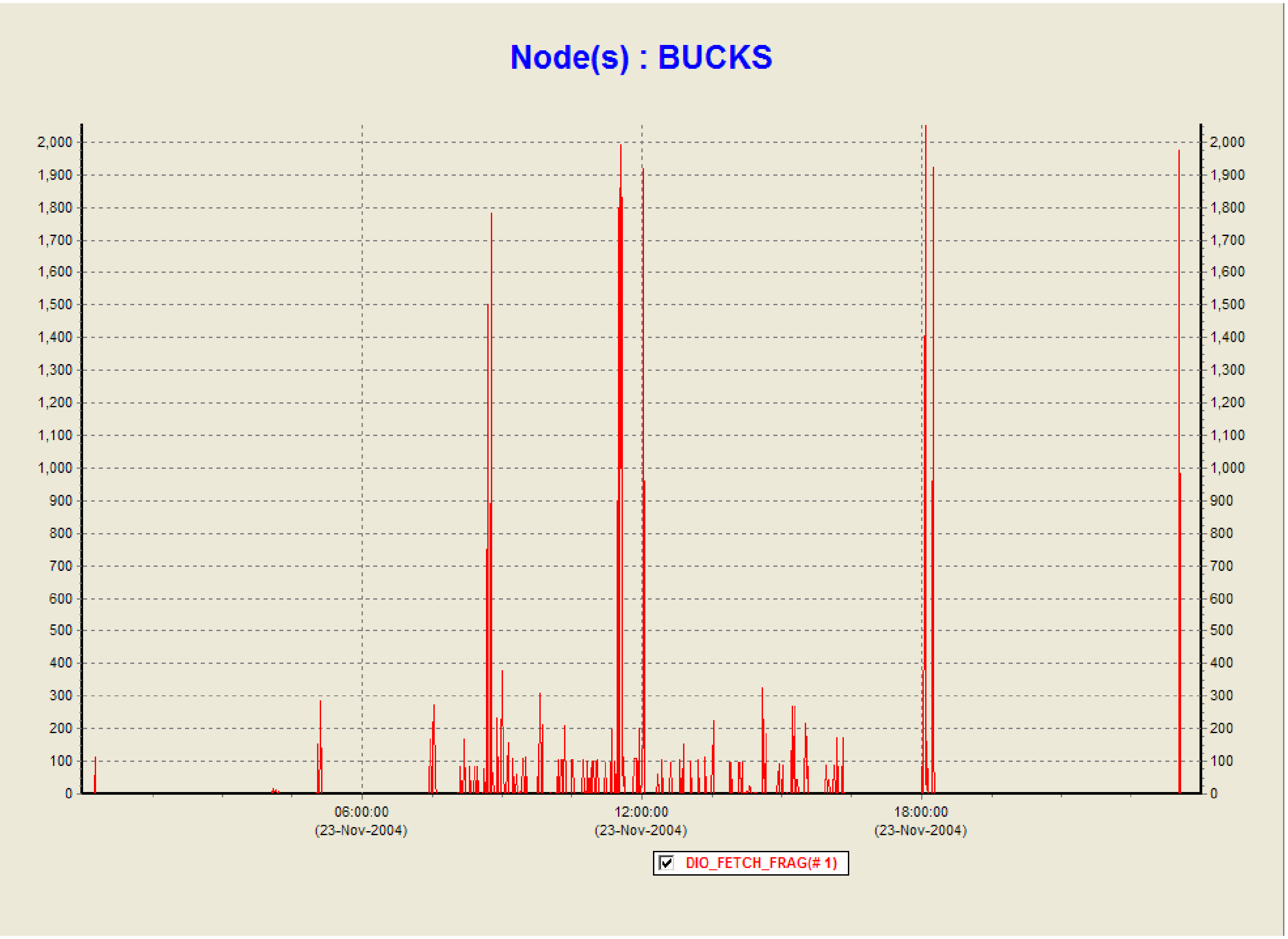
For this system and database VMS Direct I/Os match Rdb PIO_D_Notfound_IO.

- Rdb reads data pages if it cannot find them in buffers
- Spam page reads are not a significant piece of the I/O in this database
- Peaks may indicate poorly tuned queries that access lots of data

Fragmented Records – Before Reorg



Fragmented Records – After Reorg

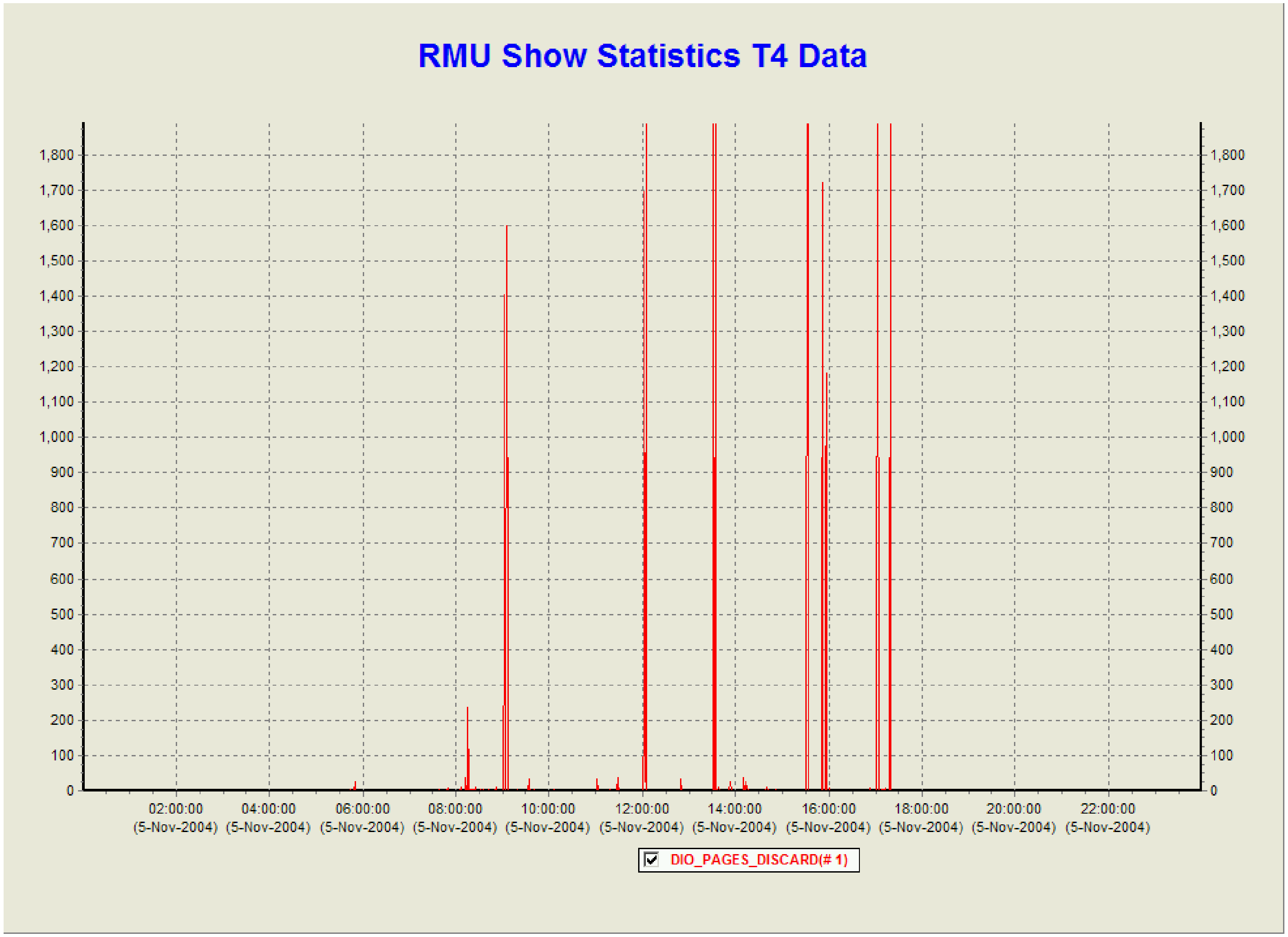


Comments on Fragmented Records

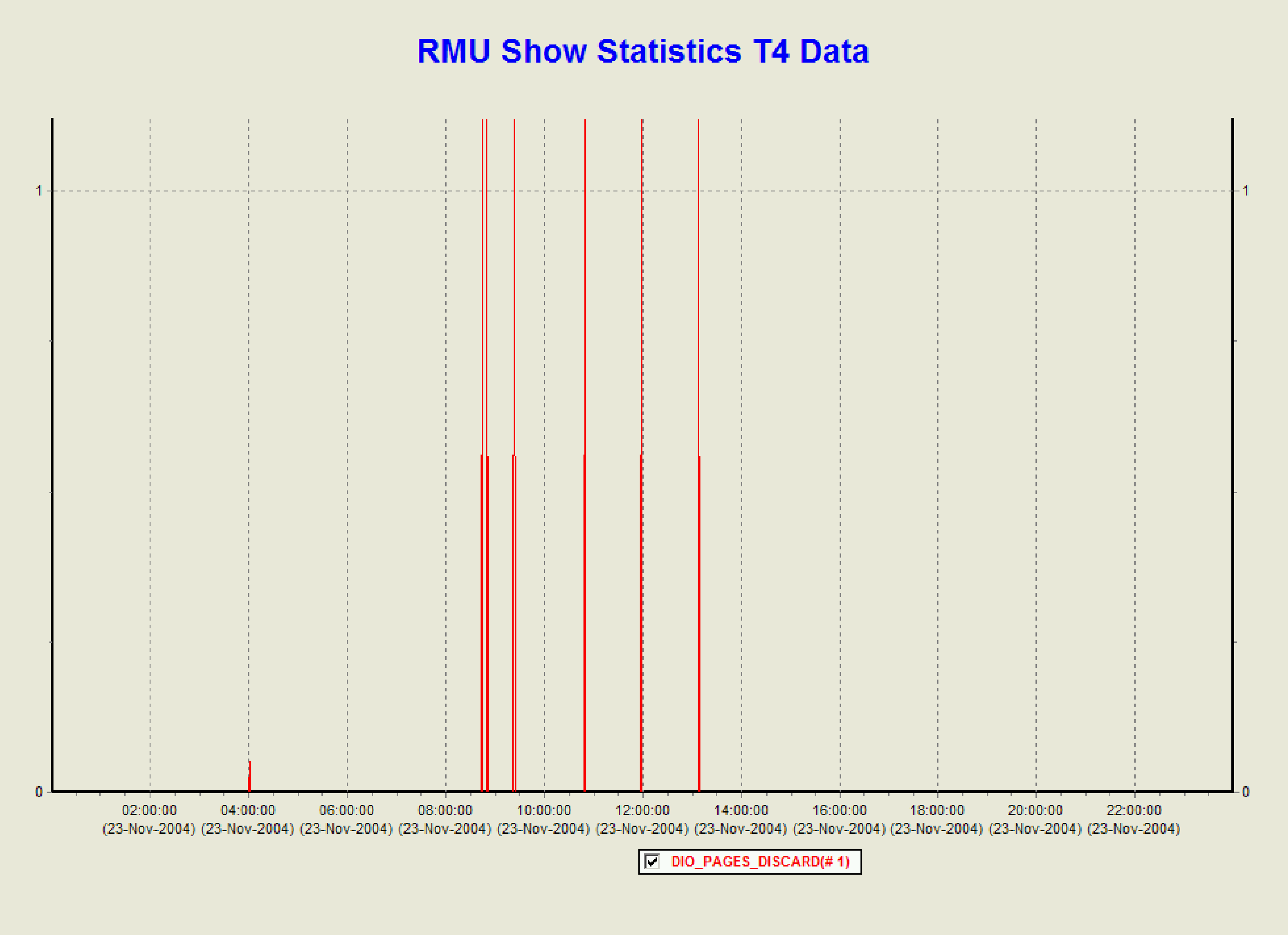
Large number of fragmented records before database reorganization.

- Some fragmented records accessed after database reorganization.

Discarded Pages – Before Reorg



Discarded Pages – After Reorg



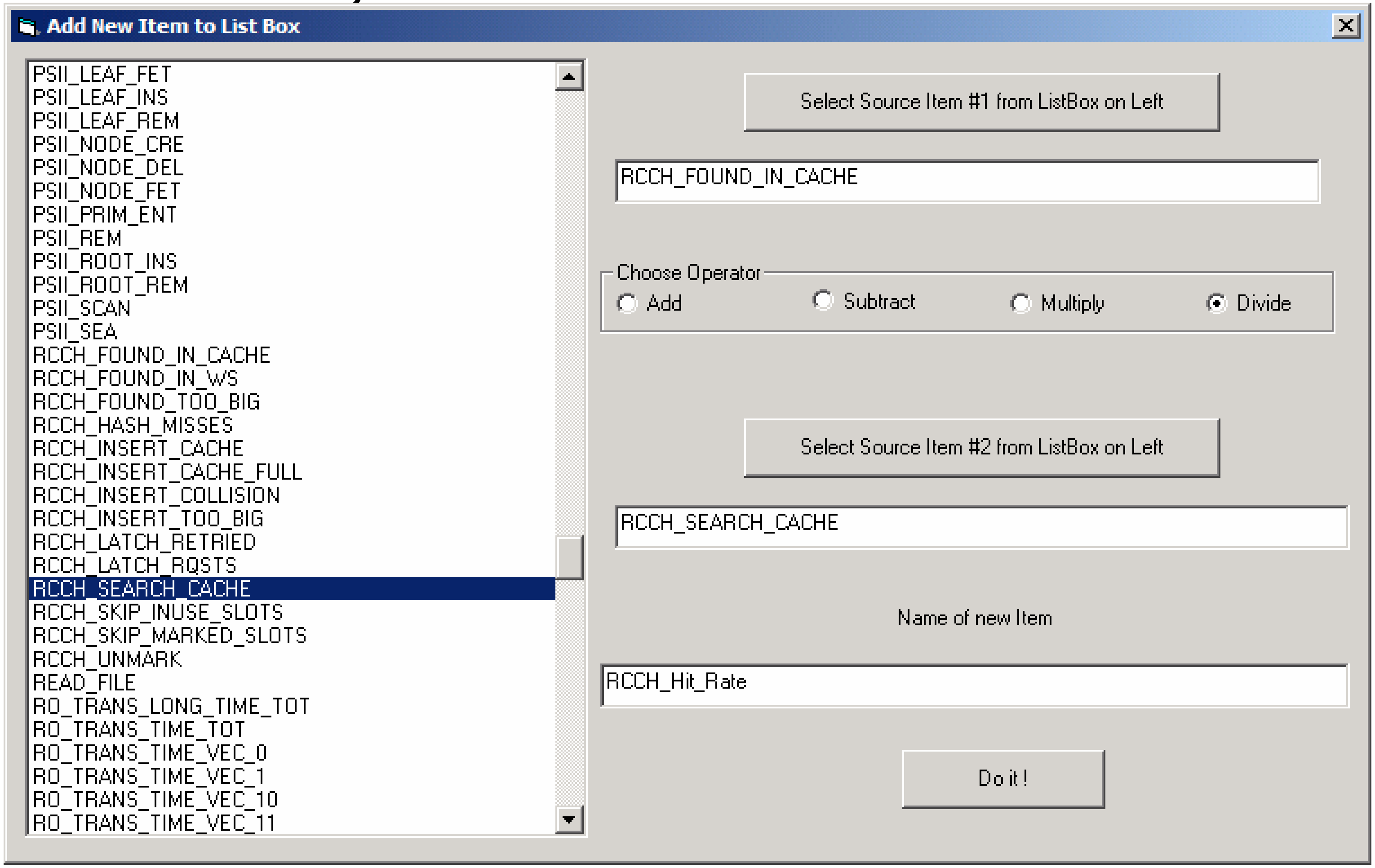
Comments on Discarded Pages

Peaks of discarded pages before database reorganization.

- Very few discarded pages after database reorganization.

Row Cache Hit Rate

TLVis allows you to create a new item:


Add New Item to List Box
Select Source Item #1 from ListBox on Left
Select Source Item #2 from ListBox on Left
Do it !

PSII_LEAF_FET
PSII_LEAF_INS
PSII_LEAF_REM
PSII_NODE_CRE
PSII_NODE_DEL
PSII_NODE_FET
PSII_PRIM_ENT
PSII_REM
PSII_ROOT_INS
PSII_ROOT_REM
PSII_SCAN
PSII_SEA
RCCH_FOUND_IN_CACHE
RCCH_FOUND_IN_WS
RCCH_FOUND_TOO_BIG
RCCH_HASH_MISSES
RCCH_INSERT_CACHE
RCCH_INSERT_CACHE_FULL
RCCH_INSERT_COLLISION
RCCH_INSERT_TOO_BIG
RCCH_LATCH_RETRIED
RCCH_LATCH_RQSTS
RCCH_SEARCH_CACHE
RCCH_SKIP_INUSE_SLOTS
RCCH_SKIP_MARKED_SLOTS
RCCH_UNMARK
READ_FILE
RO_TRANS_LONG_TIME_TOT
RO_TRANS_TIME_TOT
RO_TRANS_TIME_VEC_0
RO_TRANS_TIME_VEC_1
RO_TRANS_TIME_VEC_10
RO_TRANS_TIME_VEC_11

RCCH_FOUND_IN_CACHE

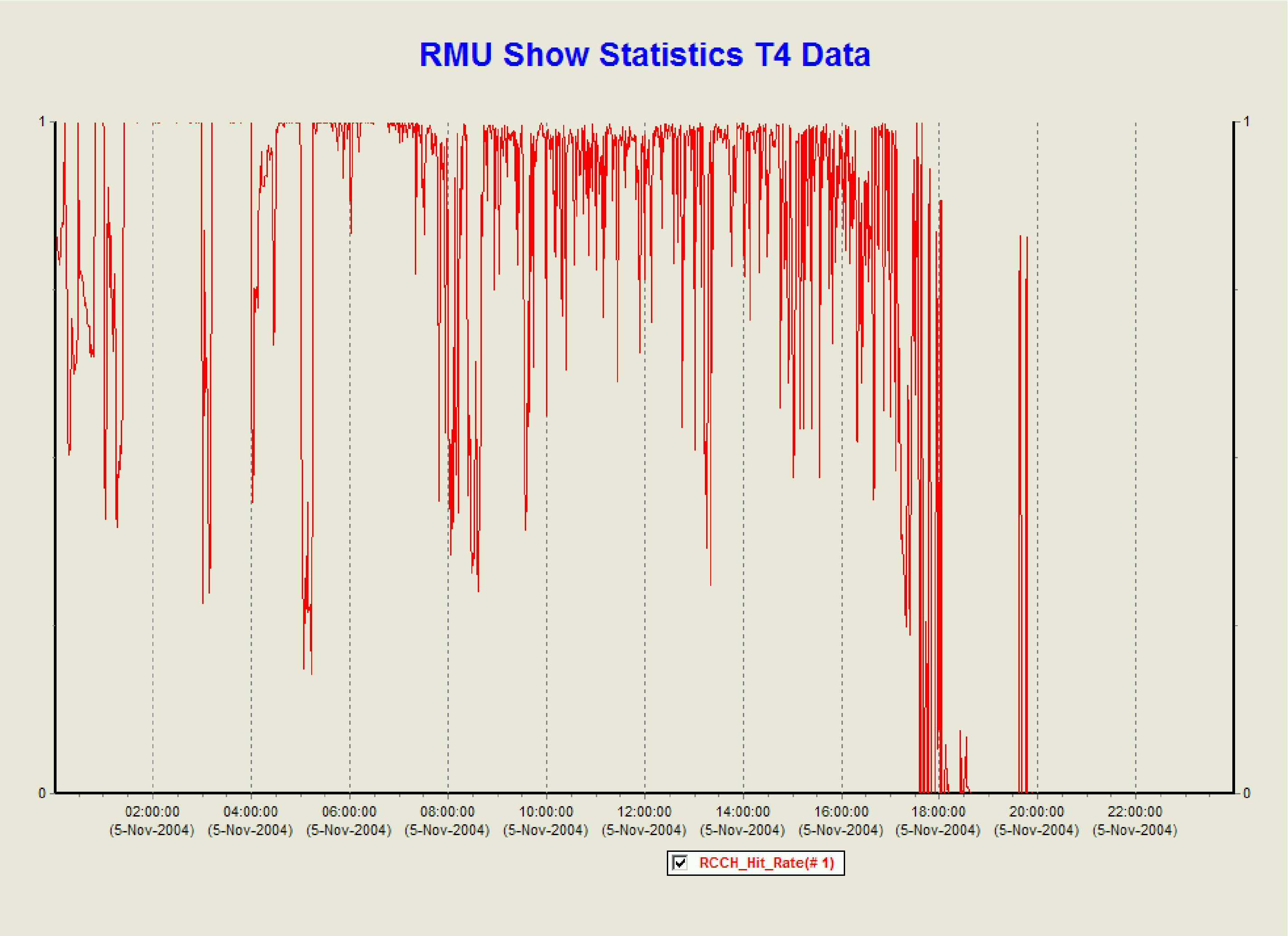
Choose Operator
○ Add ○ Subtract ○ Multiply ● Divide

RCCH_SEARCH_CACHE

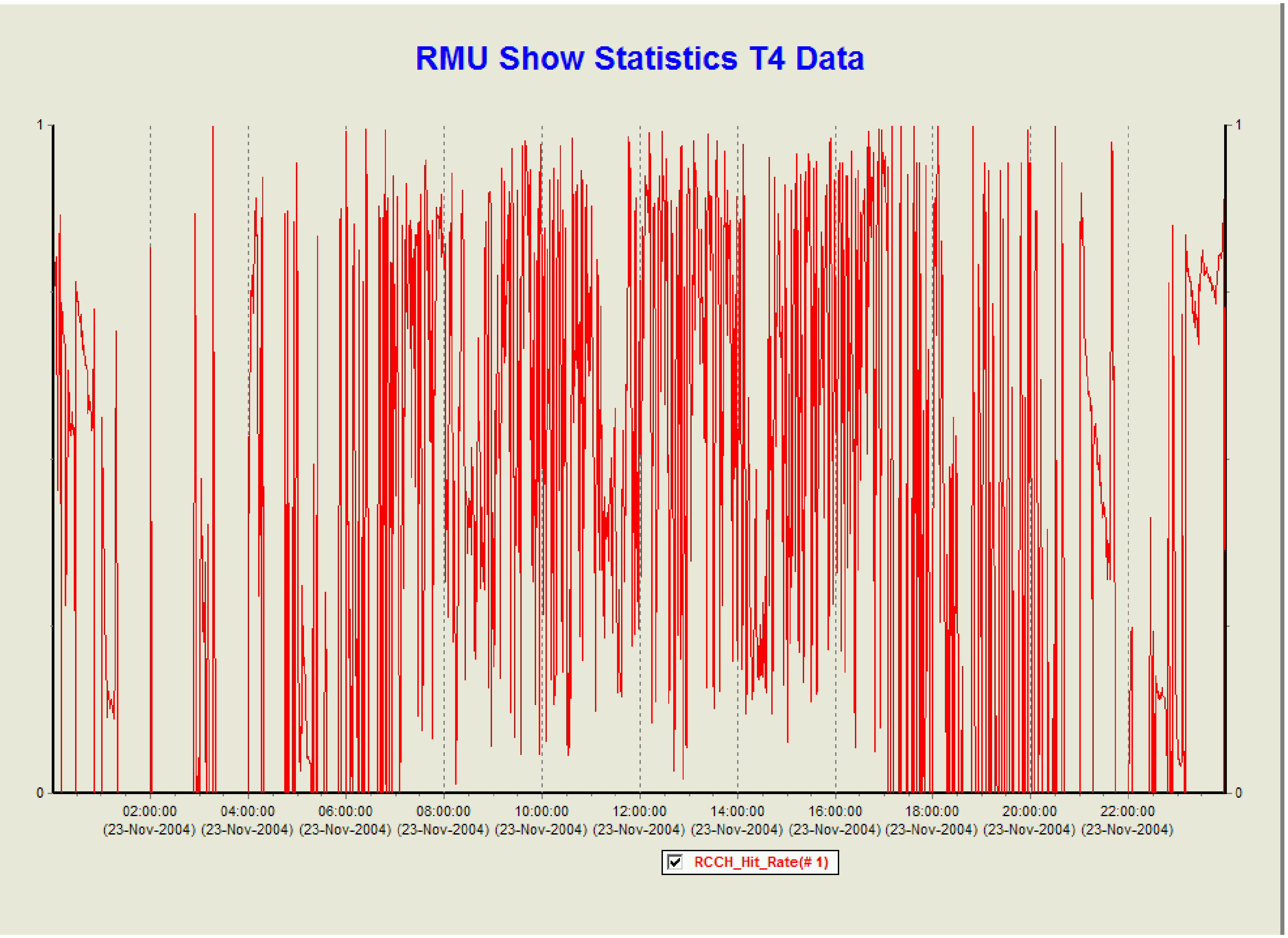
Name of new Item

RCCH_Hit_Rate

Row Cache Hit Rate: 1 Month Uptime



Row Cache Hit Rate: 1 Day Uptime

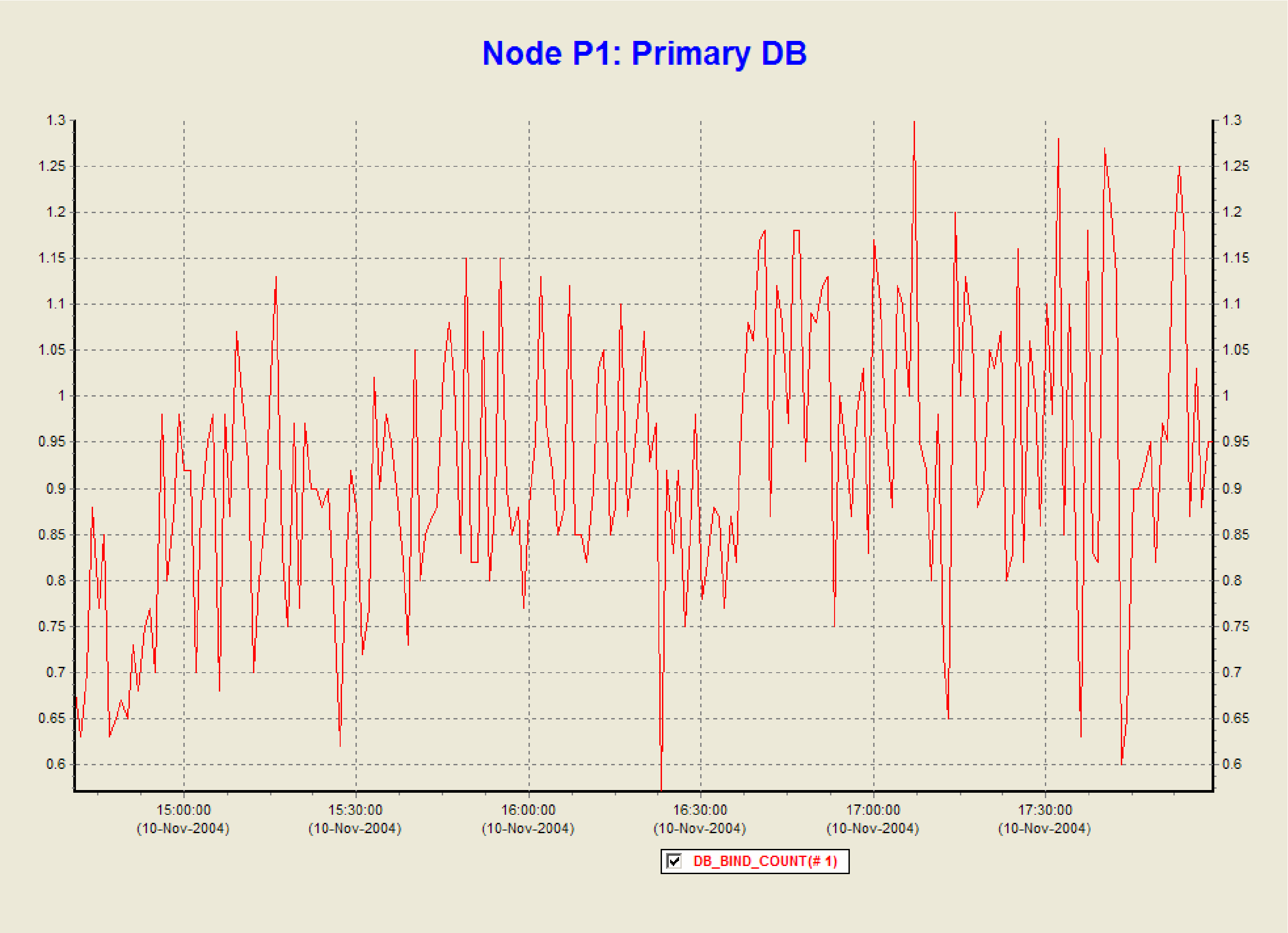


Comments on Row Cache Hit Rate

Row caches are populated over time.

- Higher hit rate the longer a database has been opened.

Frequent Attaches

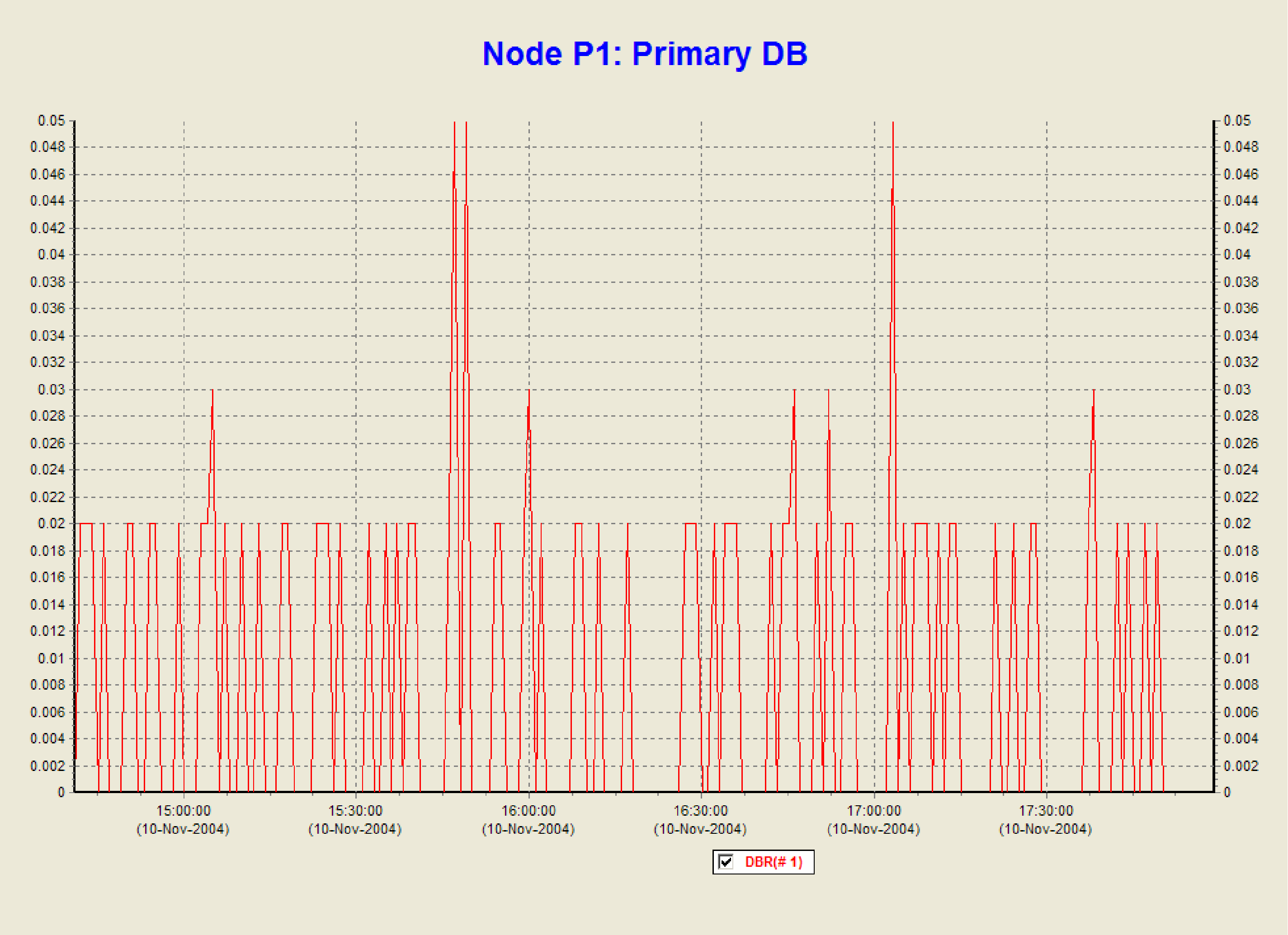


Frequent Attaches

This database is averaging .7 to .8 database attaches per second.

- Application performance problem
- Rdb$system area is set to read-only
  - If not, I/O to db root would overwhelm the system
- Issue with application design

Frequent Database Recoveries



Comments on Database Recoveries

1,000 Database Recoveries per day.

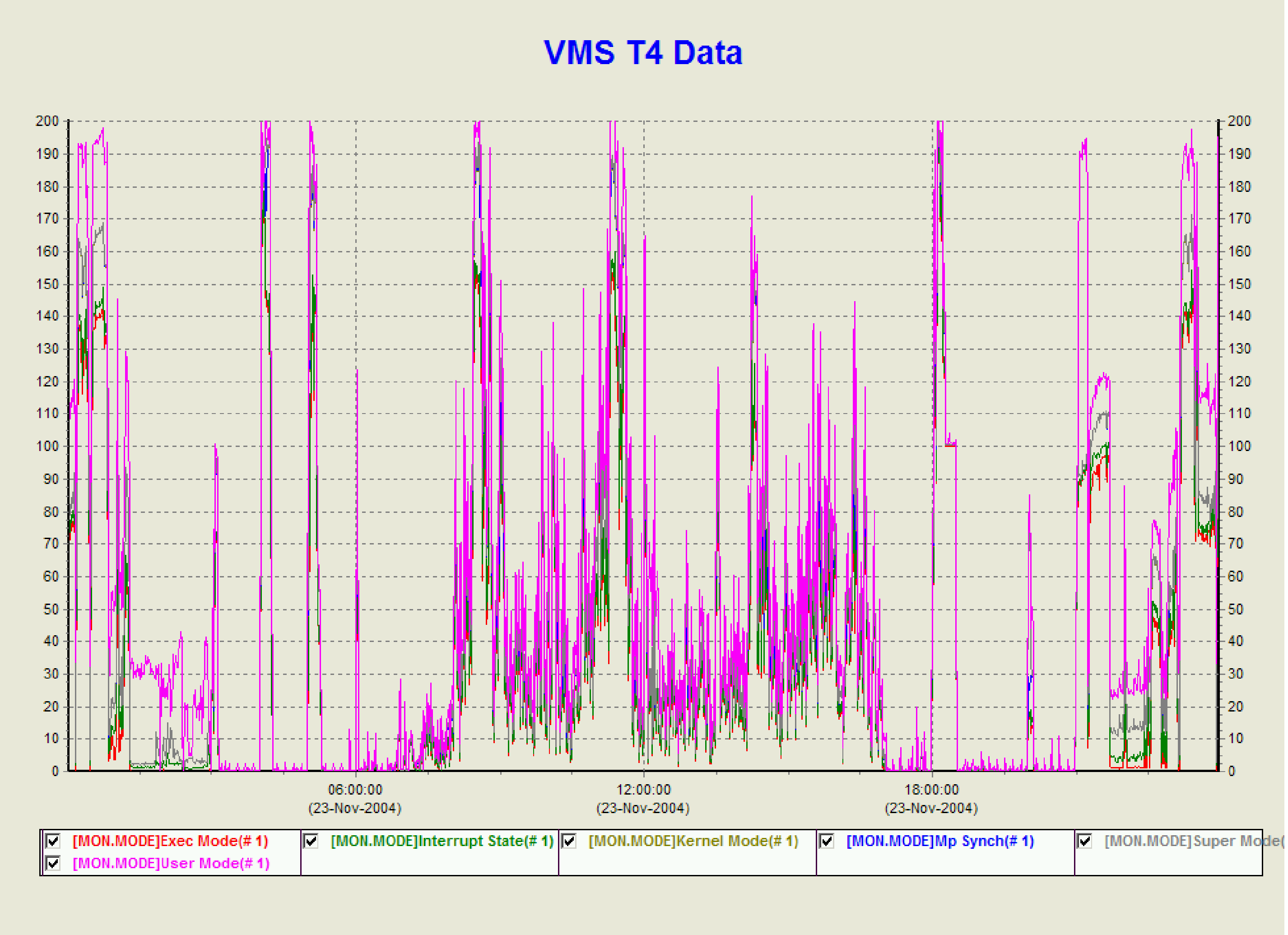
- Abnormal process terminations
- Performance issue – Freeze Locks block other processes
- Issue needs to be identified.

Predicting Future Performance Bottlenecks

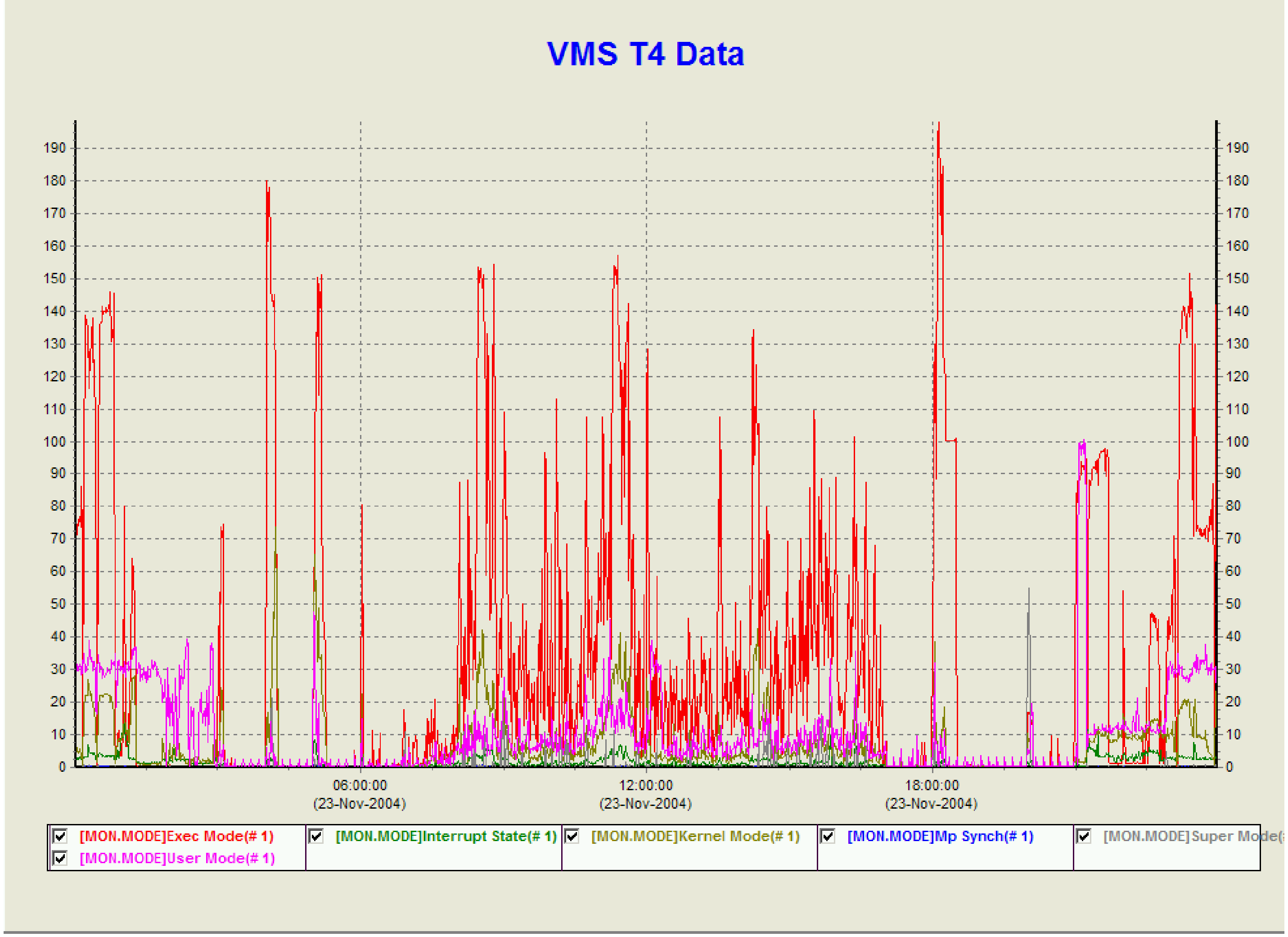
Performance bottlenecks are likely to show up in three areas:

- CPU
- I/O
- Locks

CPU Modes – Stacked



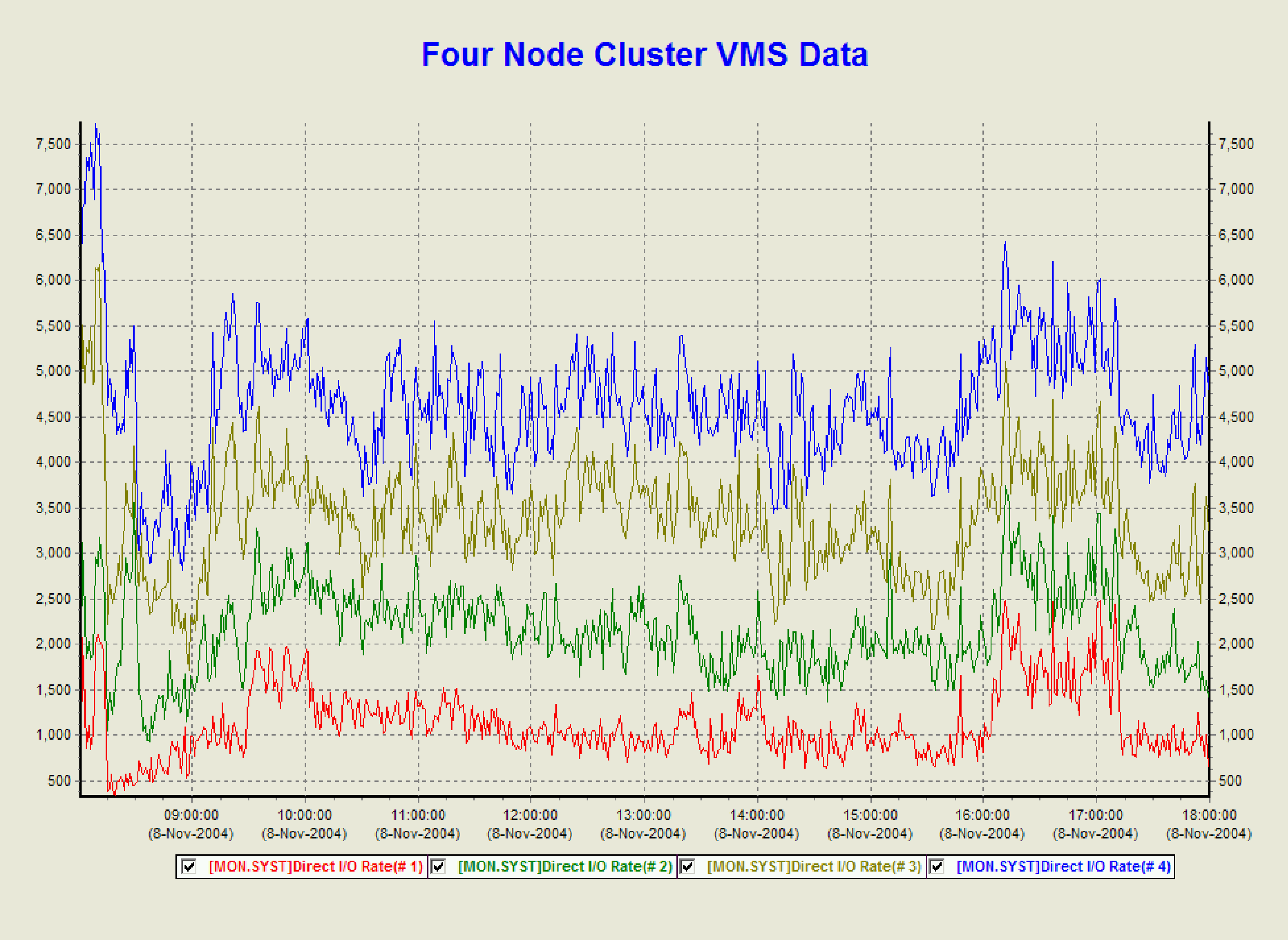
CPU Modes – Unstacked



CPU Modes – Comments

- Several points where 200% utilization of a 2-cpu system is
- Biggest piece is Exec mode – Rdb
- Options for improvement
  - Tune queries
  - More CPUs
  - Faster CPUs

I/O

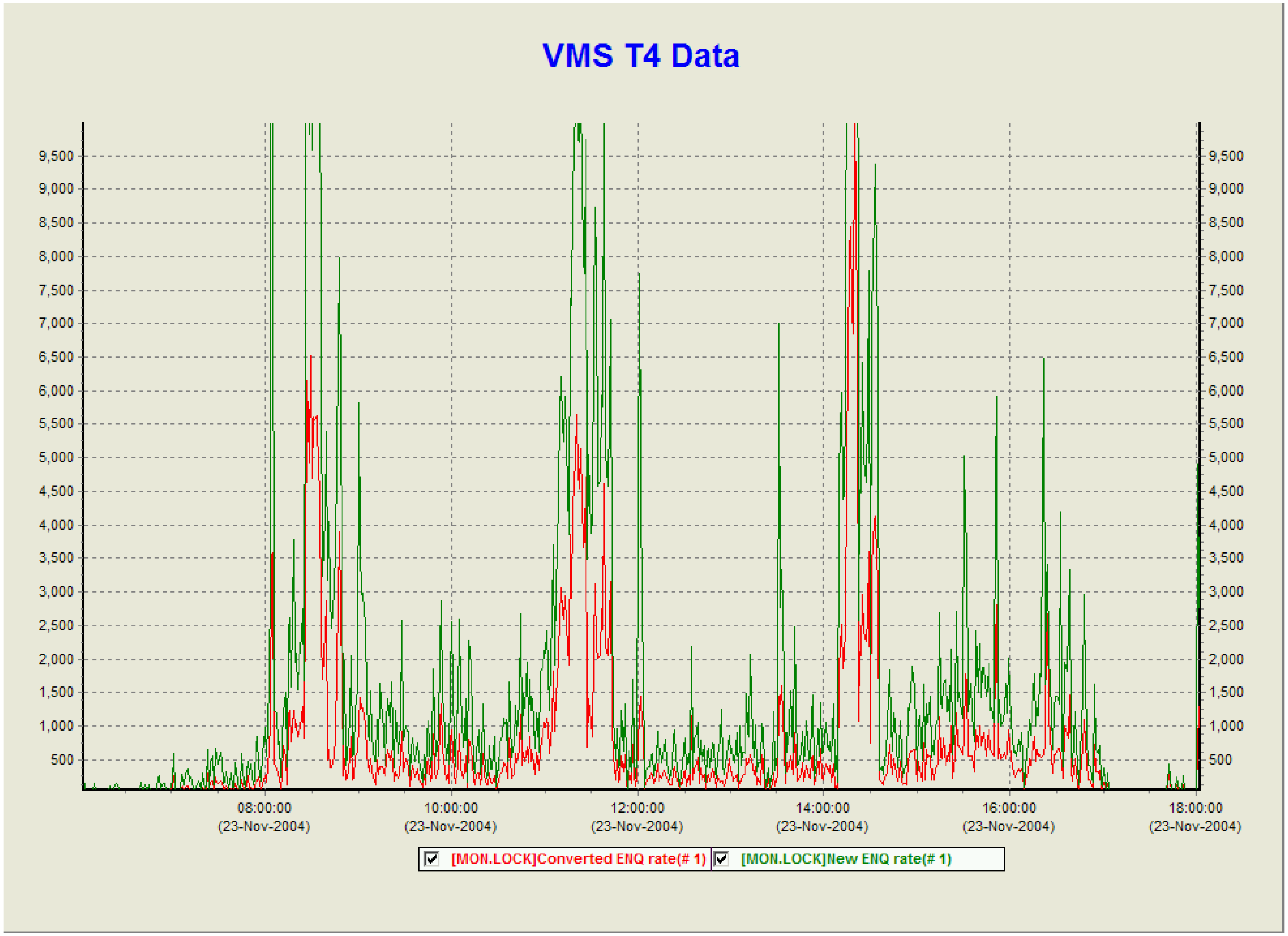

Four Node Cluster VMS Data
7,500
7,000
6,500
6,000
5,500
5,000
4,500
4,000
3,500
3,000
2,500
2,000
1,500
1,000
500
09:00:00 (8-Nov-2004)
10:00:00 (8-Nov-2004)
11:00:00 (8-Nov-2004)
12:00:00 (8-Nov-2004)
13:00:00 (8-Nov-2004)
14:00:00 (8-Nov-2004)
15:00:00 (8-Nov-2004)
16:00:00 (8-Nov-2004)
17:00:00 (8-Nov-2004)
18:00:00 (8-Nov-2004)
☑ [MON.SYST]Direct I/O Rate(# 1)
☑ [MON.SYST]Direct I/O Rate(# 2)
☑ [MON.SYST]Direct I/O Rate(# 3)
☑ [MON.SYST]Direct I/O Rate(# 4)

I/O Comments

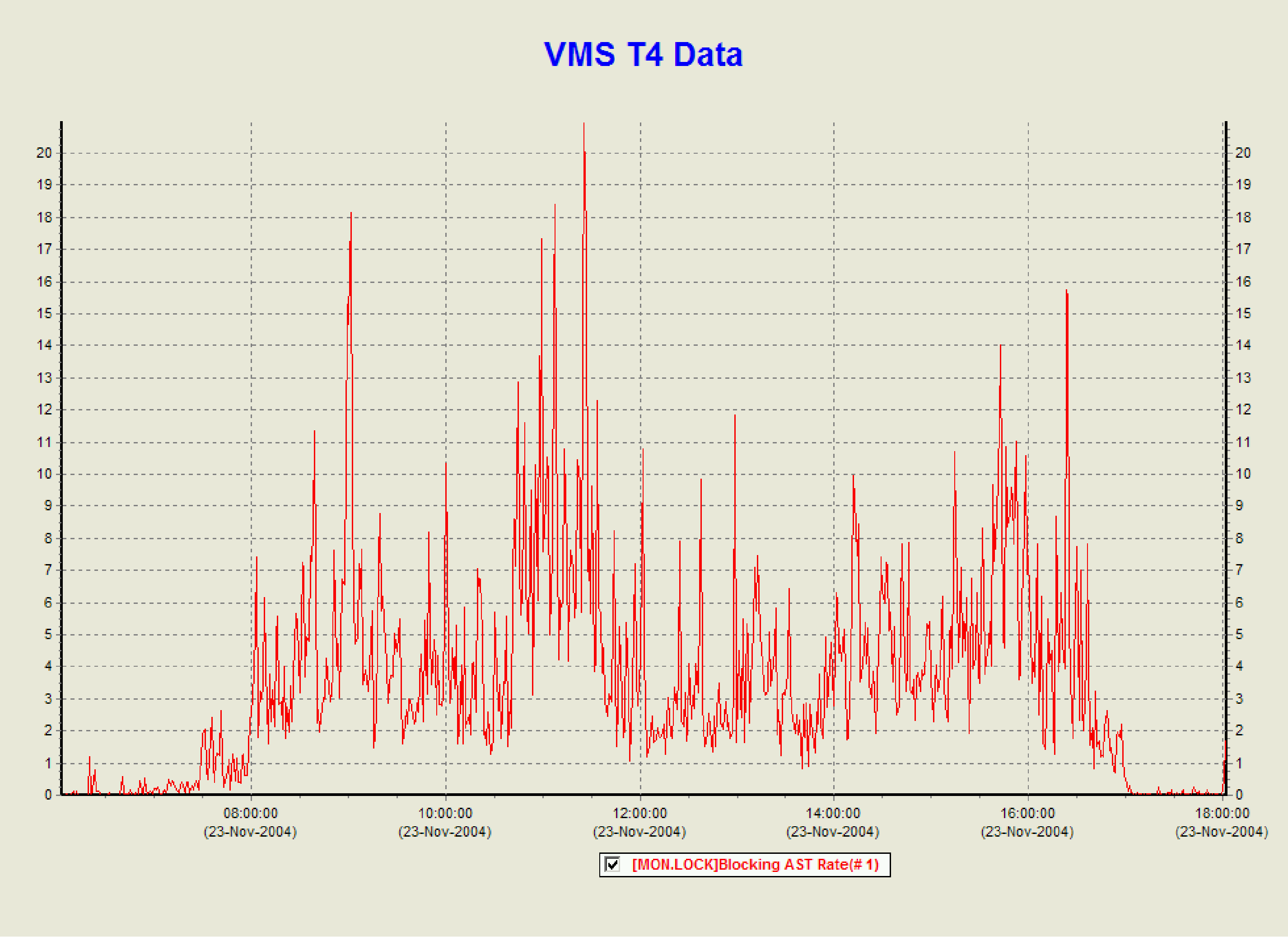
5,000 I/Os per second are possible with current I/O Subsystems.

- This system is not I/O bound – yet.
- Averaging 4,500 I/Os per second across the four nodes in the cluster is high
- Tuning is needed to reduce the I/O

Locks


VMS T4 Data
9,500
9,000
8,500
8,000
7,500
7,000
6,500
6,000
5,500
5,000
4,500
4,000
3,500
3,000
2,500
2,000
1,500
1,000
500
08:00:00 (23-Nov-2004)
10:00:00 (23-Nov-2004)
12:00:00 (23-Nov-2004)
14:00:00 (23-Nov-2004)
16:00:00 (23-Nov-2004)
18:00:00 (23-Nov-2004)
☑ [MON.LOCK]Converted ENQ rate(# 1)
☑ [MON.LOCK]New ENQ rate(# 1)

Blocking AST



Comments On Locks and Blocking ASTs

- Wide range of lock operations suggest that there is no problem
- Flattened Peaks would indicate bottlenecks
- Blocking ASTs indicate some page contention

Additional Opportunities

- T4 format data could be loaded into a an Rdb database
- Use SQL to identify long term trends

Conclusions

- T4 is a low-impact set of tools for gathering VMS performance data
- RMU Show Statistics information binary output files can be converted to T4-format CSV files
- The combination of information is very useful for monitoring performance.

Questions?