Scheduler and Dispatcher

Velocity Software Inc.
196-D Castro Street
Mountain View CA 94041
650-964-8867

Velocity Software GmbH
Max-Joseph-Str. 5
D-68167 Mannheim
Germany
+49 (0)621 373844

Objectives

- Understanding Scheduler / Dispatcher
- How SRM affects users
- How SHAREs affect users

What is important?

- Running at HIGH Utilization!!!
- When users / servers get dispatched
  - Prioritizing work (Share values)
- How long are they dispatched for (time slice)
- What happens when there are resource constraints (eligible list)

Starting with 3 looping users RELATIVE 100 share

- They all get equal share of the resources
- this is as we expected.

```
Screen: ESAUSP2  Velocity Software-Test VSIVM4  ESAMON 3.778
 1 of 3  User Percent Utilization                CLASS * USER
                              <-------Main Storage-------->
         UserID   <Processor> <Resident->  Lock <-WSSize-->
Time     /Class   Total  Virt Total  Actv   -ed Total  Actv
-------- -------- ----- ----- ----- ----- ----- ----- -----
00:11:00 ROBLNX1  32.39 32.38 15862 15862    11 15536 15536
         ROBLX2   32.12 32.11 66136 66136   259 78478 78478
         ROBLX1   32.02 32.01 38219 38219   176 37790 37790
         ROB2LV    0.01  0.00  2246  2246     0  2246  2246
```

Because it doesn’t work the way you expect

We now give ROBLX2 a RELATIVE 200 share

- because that is a more important service
- (nothing with virtual 2-way).
- Not as expected, it gets the excess share

```
 Screen: ESAUSP2  Velocity Software-Test VSIVM4  ESAMON 3.778
1 of 3  User Percent Utilization                CLASS * USER
                              <--------Main Storage-------->
         UserID   <Processor> <Resident->  Lock <-WSSize-->
Time     /Class   Total  Virt Total  Actv   -ed Total  Actv
-------- -------- ----- ----- ----- ----- ----- ----- -----
00:14:00 ROBLX2   68.71 68.68 66211 66211   258 78478 78478
         ROBLX1   14.00 14.00 38245 38245   256 37790 37790
         ROBLNX1  13.99 13.99 15879 15879    11 15536 15536
         ROB2LV    0.01  0.00  2246  2246     0  2246  2246
```

The Scheduler

- Maintains the lists of users
  - Eligible, Dispatch, Dormant
- Calculates “deadline” priorities
- Determines Eligibility to be Dispatchable

The Dispatcher

- Selects a user to run
- Dispatches units of work

Command Overview

Scheduler affected by:

- SET SRM STORBUF (control storage utilization)
- SET SRM DSPBUF (control processor utilization)
- SET SRM LDUBUF (control paging device utilization)
- SET SRM DSPSLICE (time slize, default 5ms)
- SET SRM IABIAS (bias interactive users)
- **SET SHARE (guarantee a share of CPU)**
- SET QUICKDSP (ignore STORBUF, DSPBUF, LDUBUF)

Dispatcher affected by:

- SET SRM DSPSLICE

SET SHARE

Shares are “normalized” to workload

- Absolute is fixed percent
- Relative is relative to other relative

Absolute vs Relative

- Absolute shares go up as workload increases
- Relative shares go down as workload increases

Use Absolute shares for: (Ignore IBM defaults)

- **Servers that need more resource as more users log on**
- **Examples: TCPIP, RACF, Database servers**

Use Relative shares for users

QUICKDSP does NOT impact share values!

Scheduler Lists Overview

Dormant List

- Idle users, those logging on, logging off
- No special order
- Any user idle for 300ms or more,
- Traditional CMS workloads

Eligible List (mostly not used anymore)

- Contains users who want to consume resources
- Users not yet allowed to contend,
  - Short on storage
  - Short on paging devices
- Kept in priority order

Dispatch List

- Users contending for resources now
- Kept in priority order
- Linux always here

VELOCITY SOFTWARE

Important Terms

Dispatch Queue (Dispatch List)

- The list of virtual machines requesting resource (working)

Dispatch Time Slice

- maximum time virtual machine dispatched

Elapsed time slice

- Maximum Time in queue before q-drop

Queue Drop (Prior to z/vm 6.3)

- virtual machine is done working, or ETS has expired

Dormant List

- Idle users **(Idle for 300ms)**

Eligible List

- Virtual machines that want to do work, but are held back

Dispatch List Definition

Class 1 (Interactive)

- Entry from the Dormant List
- Initial Q1ETS (variable from .05 seconds to 16 seconds)
- IA (InterActive) Bias applies

Class 2 (Non–Interactive)

- Entry after one ETS in Class 1
- Q2ETS is 8x Class 1 ETS (fixed multiple)
- Long running user will get 1 Q2ETS stay in Q2 before demotion

Class 3 (Long–running, batch, guests)

- Entry after one stay (8x ETS) in Class 2
- Q3ETS is 48x the Class 1 ETS (fixed multiple)

Objective: Give trivial interactive transactions priority

Users start in class 1, graduate to class 2, then 3

Dispatch List Workloads

Class 1 (Interactive)

- CMS Users
- Idle Linux users with timer patch

Class 2 (Non–Interactive)

- Long running CMS users

Class 3 (Long–running, batch, guests)

- Z/OS, TPF
- Idle Linux
- Active Linux guest
- WAS, Domino, SAP servers

Class 0 (No eligible list, treated as Class 2)

- Hot shot, Lock shot users

Example, Linux users in Queue 3

```
Report: ESAUSRQ      User Queue and Load Analysis
------------------------------------------------------------------------
         <-----------User Load------------>      <-----------Average Num
UserID   Logged  Non-          Disc- Total  Tran <-------Dispatch List--
/Class       on  Idle   Active conn  InQue  /min    Q0    Q1    Q2    Q3
--------- ------ ----- ------- ------ ----- ---- ----- ----- ----- -----
05:06:00   58.0     .     33.2      .  25.4   259   4.0   2.4   0.6  18.4
Hi-Freq:   58.0    34     33.2     56  23.7   233   3.3   0.6   1.5  18.3
 ***Key User Analysis ***
VMSECURE    1.0     1      1.0      1     0   3.6     0     0     0     0

 ***User Class Analysis***
Servers    16.0     9      9.0     14   0.1  20.0     0   0.1     0     0
KeyUsrs     2.0     2      2.0      2   1.3   106   1.3     0     0     0
ZVPS        9.0     5      5.0      9   0.1  37.2     0   0.1     0     0
Linux      13.0    12     12.0     13  20.1  35.6     0   0.3   1.5  18.3
TheUsers   15.0     4      3.2     15   2.0  30.4   2.0   0.0     0     0

***Top User Analysis***
ZLNXB20     1.0     1      1.0      1   1.0     0     0     0     0   1.0
ZLNXB15     1.0     1      1.0      1   1.0     0     0     0     0   1.0
ZLNXB21     1.0     1      1.0      1   1.0     0     0     0     0   1.0
ZLNXB16     1.0     1      1.0      1   1.0     0     0     0     0   1.0
ZLNXB17     1.0     1      1.0      1   1.0     0     0     0     0   1.0
ZLNXB10     1.0     1      1.0      1   1.0   9.6     0   0.1   0.4   0.5
ZLNXB18     1.0     1      1.0      1   1.0     0     0     0     0   1.0
```

High Level Scheduler

Fair Share Scheduler (Wheeler scheduler):

- Allows prioritization of work
- Determines work “Eligibility”
- Protects workload from resource over commitment using the “eligible List” - no “Thrashing”
- Supports 1000’s of concurrent virtual machines
- Maintains dispatch list to create fair share
- Allows wide range of workloads to effectively utilize resource

Also called DEADLINE SCHEDULING

- Every inqueue user assigned a deadline

Question: What are we trying to control with Eligible?

- Fair share based on business requirements
- System responsiveness when resources constrained

The Death Spiral

The Death Spiral: If service of service machine (or other resource) is slow, then:

1. Users are delayed and transaction time increases
2. Storage requirements increase (because there is more concurrent workload)
3. Paging requirements increase
4. Go to 1.

Setting Shares Not changed since VM/XA

Looping users (1991 survey done with vtam)

- Does a looping user affect other users?
- Do you have TCPIP at relative share 10000?
- Are TCPIP's high share and looping users affecting other users related?
- How much excess share does RELATIVE 10000 create?

Why set share to relative 10000 anyway???

- Recommendation from VM development without analysis? They don’t recommend it now.
- Destroys scheduler ability to “fair share”

What is normalized share?

Calculation of Normalized Share

All ABSOLUTE and RELATIVE shares “normalized”

- Sum the Absolute shares of all VMDBKs in Dispatch list (SRMABSDL)
- Sum the Relative shares of all VMDBKs in Dispatch List (SRMRELDL)

```
Report: ESASUM System Summary
Variable Average Minimum Maximum Description
-------- ------- ------- ------- -----------------------------------
SRMBIASI      90                 Interactive bias intensity percent (SET SRM I
SRMBIASD       2                 Interactive bias duration (SET SRM IAB)

SRMTSLIC    5.00                 Minor time slice (ms) (SET SRM DSPSLICE)
SRMTSHOT    2.00                 Minor time slice (ms) for HOTSHOT users

SRMABSDL    52.0    48.0    55.0 Total absolute shares of VMDBKs in the dispat
SRMRELDL     818     550    1900 Total relative shares of VMDBKs in the dispat
```

Calculation of Normalized Share

If SRMABSDL is less than 100%

- Normalized share equals Absolute Share
- Relative Share users get:

(100 - SRMABSDL) x (relative share / SRMRELDL)

If SRMABSDL is greater than 99,

- Absolute shares “normalized” to 99
- Relative users “share” 1 percent
- Very dangerous situation

Normalized shares are percentages of the CPU resource

Delay factor (OFFSET) is then DSPSLICE / “normalized” share

Excess Share Analysis Pre 6.4

Starting with 3 looping users RELATIVE 100 share

- They all get equal share of the resources
- this is as we expected.

```
Screen: ESAUSP2  Velocity Software-Test VSIVM4  ESAMON 3.778
 1 of 3  User Percent Utilization                CLASS * USER
                              <-------Main Storage-------->
         UserID   <Processor> <Resident->  Lock <-WSSize-->
Time     /Class   Total  Virt Total  Actv   -ed Total  Actv
-------- -------- ----- ----- ----- ----- ----- ----- -----
00:11:00 ROBLNX1  32.39 32.38 15862 15862    11 15536 15536
         ROBLX2   32.12 32.11 66136 66136   259 78478 78478
         ROBLX1   32.02 32.01 38219 38219   176 37790 37790
         ROB2LV    0.01  0.00  2246  2246     0  2246  2246
```

Excess Share Analysis

We now give ROBLX2 a RELATIVE 200 share

- because that is a more important service
- (nothing with virtual 2-way).
- Not as expected, it gets the excess share

```
 Screen: ESAUSP2  Velocity Software-Test VSIVM4  ESAMON 3.778
1 of 3  User Percent Utilization                CLASS * USER
                              <-------Main Storage-------->
         UserID   <Processor> <Resident-> Lock <-WSSize-->
Time     /Class   Total  Virt Total  Actv  -ed Total  Actv
-------- -------- ----- ----- ----- ----- ----- ----- -----
00:14:00 ROBLX2   68.71 68.68 66211 66211  258 78478 78478
         ROBLX1   14.00 14.00 38245 38245  256 37790 37790
         ROBLNX1  13.99 13.99 15879 15879   11 15536 15536
         ROB2LV    0.01  0.00  2246  2246    0  2246  2246
```

Excess Share Analysis

Now for the experiment – Set shares “correctly”

- we reduce the relative share for all idle but inqueue users down to 1
- Convert TCPIP from REL 3000 to ABS 2%
- (using the allocated share computation below and showing how much allocated / consumed share is).
- This ELIMINATES “EXCESS” bucket – allows perfect case scenario

```
Screen: ESAUSP2  Velocity Software-Test VSIVM4  ESAMON 3.778
1 of 3  User Percent Utilization                CLASS * USER
                              <--------Main Storage-------->
         UserID   <Processor> <Resident->  Lock <-WSSize-->
Time     /Class   Total  Virt Total  Actv   -ed Total  Actv
-------- -------- ----- ----- ----- ----- ----- ----- -----
00:20:00 ROBLX2   48.39 48.37 67141 67141   292 80047 80047
         ROBLNX1  24.19 24.19 16168 16168    11 15536 15536
         ROBLX1   24.19 24.18 39006 39006   241 37790 37790
         ROB2LV    0.01  0.00  2246  2246     0  2246  2246
```

Excess Share Analysis (6.4)

Starting with 3 looping users RELATIVE 100 share

- They all get equal share of the resources
- this is as we expected.

```
Screen: SMART    Velocity Software        ESAMON 4.301 01/22 09:47-09:
1 of 1  Smart
-----------------------------------  ---------------------------------
    <----------Top Users---------->  <-----------Servers------------>
    Userid:     CPU%  IO/Sec Pg/Sec  Userid:     CPU%   IO/Sec Pg/Sec
 1) BART2       27.8        0      0  System:     88.0    16.12      0
 2) BART3       27.8     0.33      0  RACFVM       0.2     1.95      0
 3) BART1       27.2        0      0  TCPIP        0.2        0      0
 4) OPERATOR     1.1        0      0  TCPIP2       0.1        0      0
 5) ZVPS         1.1        0      0  RSCS         0.0        0      0
 7) VMSYSVPS     0.8    12.47      0
10) ZWRITE       0.3     1.00      0
```

Excess Share Analysis (6.4)

We now give BART2 a RELATIVE 200 share

- because that is a more important service
- Not as expected, And low excess share, not as expected

```
Screen: SMART    Velocity Software        ESAMON 4.301 01/22 09:53
1 of 1  Smart
----------------------------------  ------------------------------------
    <----------Top Users-----------> <-----------Servers------------->
    Userid:     CPU%  IO/Sec Pg/Sec  Userid:     CPU%   IO/Sec Pg/Sec
 1) BART2       48.9       0      0  System:     89.9     1.00      0
 2) BART1       19.3       0      0  TCPIP        0.2        0      0
 3) BART3       19.3       0      0  TCPIP2       0.1        0      0
 5) ZWRITE       0.3    0.50      0
 7) ZVPS         0.2       0      0
 8) ZTCP         0.1       0      0
 9) VMSYSVPS     0.0    0.38      0
```

Excess Share Analysis

Share settings:

- BART1: 100, BART2:   200,   BART3:   300
- Not as expected, Low excess share, TCPIP ABS 3%

```
 Screen: SMART    Velocity Software          ESAMON 4.301 01/22 09:5
1 of 1  Smart
-----------------------------------  -----------------------------------
    <----------Top Users---------->  <-----------Servers------------>
    Userid:     CPU%  IO/Sec Pg/Sec  Userid:     CPU%   IO/Sec Pg/Sec
 1) BART3       45.0       0      0  System:     88.0    16.15      0
 2) BART2       29.0       0      0  RACFVM       0.2     1.40      0
 3) BART1        9.9       0      0  TCPIP        0.2        0      0
 4) OPERATOR     0.8       0      0  TCPIP2       0.2        0      0
 7) VMSYSVPS     0.7   12.87      0
 9) ZWRITE       0.3    0.77      0
10) ZVPS         0.2       0      0
```

Excess Share Analysis (6.4)

Share settings – WITH EXCESS SHARE 10000:

- BART1: 100, BART2:   200, BLAKE001: 10000
- Almost looks right (Did z/VM 6.4 scheduler fix it?)

```
 Screen: SMART    Velocity Software 1 of 1
-----------------------------------
    <----------Top Users---------->
    Userid:      CPU%  IO/Sec Pg/Sec
 1) BART2        52.7       0      0
 2) BART1        25.6       0      0
 3) BLAKE001      6.4    0.13      0
 4) OPERATOR      0.7       0      0
 7) VMSYSVPS      0.7   13.35      0
 8) ZWEB02        0.7    1.95      0
 9) ZWRITE        0.3    0.65      0
10) ZVPS          0.2       0      0
```

Excess Share Analysis (6.4)

Share settings – WITH EXCESS SHARE 10000:

- Everything looks very reasonable
- Did z/VM 6.4 scheduler fix it?

```
 Screen: SMART    Velocity Software 1 of 1
-----------------------------------
    <----------Top Users---------->
    Userid:     CPU%  IO/Sec Pg/Sec
 1) BART3       40.8       0      0    REL 200
 2) BART2       19.7       0      0    REL 100
 3) BART1       19.4       0      0    REL 100
 4) BLAKE001     6.8    0.17      0    REL 10000 – EXCESSS SHARE
 5) ZALERT       0.9       0      0
 9) VMSYSVPS     0.1    1.47      0
10) ZTCP         0.0       0      0
```

Excess Share Analysis (6.4)

Share settings – WITH EXCESS SHARE 10000:

- Doesn’t look right (But better than z/VM 6.3)
- Not different from when low excess share

```
 Screen: SMART    Velocity Software
-----------------------------------
    <----------Top Users---------->
    Userid:      CPU%  IO/Sec Pg/Sec
 1) BART3        41.5       0      0     REL SHARE 300   REASONABLE
 2) BART2        27.2       0      0     REL SHARE 200   REASONABLE
 3) BART1         9.8       0      0     REL SHARE 100   NOT RIGHT
 4) BLAKE001      6.8    0.47      0     REL SHARE 10000, excess
 5) ZALERT        0.8       0      0
 6) ZWRITE        0.6    5.43      0
 9) ZSERVE        0.1    0.07      0
10) ZTCP          0.1       0      0
```

Dispatch List Deadline Priority

Deadline priority is a “target” time of day

- Deadline = TOD + DelayFactor
- “Dispatch List” and “Eligible List” priority are of this type
- Based on ATOD (artificial time of day)

Dispatch list delay factor:

- Based on “Normalized” share
- Delay factor = DSPSLICE / (ncpus * normalized share)
  - 1% share will have 100 time slice delay (500ms)
- Subtract IABias (Interactive Bias – first n times enters Q1)
- Subtract PageBias (E2/E3 users with stolen pages)
- Deadline is calculated after every dispatch time slice is completed.

Setting Deadline

Scheduler builds ordered dispatch list based on deadline

Deadline time of day = current TOD + offset

Offset = (DSPSLICE / Normalized share) * bias

```
|---|---|---|---|---|---|---|---|---|---|---|---|--->(time)
(ATOD)
                                    users
                                   ||||||||||
---|---|---|---|---|---|---|---|---|---|---|---|---> (time)

TCPIP                              users
    ||                              ||||||||||
---|---|---|---|---|---|---|---|---|---|---|---|---> (time)
```

Dispatcher takes users in order from sorted deadline list

SHARE Impact on CPU Delivery Rate

CPU Delivery Rate for “one cpu system”

If normal share is 10%, user will have:

- Delivery rate = 1 dispatch time slice out of 10.
- Offset = 10 dispatch time slices.

If normal share is 50%, user will have:

- Delivery rate = 1 dispatch time slice out of 2.
- Offset = 2 dispatch time slices.

If normal share is 1%, user will have:

- Delivery rate = 1 dispatch time slice out of 100.
- Offset = 100 dispatch time slices.

Worst case seen – offset for general users:

- 30 minutes

Sample Deadlines

Example (50 users using IBM Defaults)

- RACF has relative share 10000
- TCPIP has relative share 10000
- User has relative share 100
- DSPSLICE = 5ms
- SRMRELDL = 25000 (typical)
- **(100 - SRMABSDL) x (relative share / SRMRELDL)**

Normalized share = 100 / 25000 = .004 (.4%)

- CPU Delivery rate = 5ms / .004
- = 5ms per 1.25 seconds
- Subsecond obviously NOT the design point

Sample Deadlines - Comparison

Example 1:

- TCPIP offset 2.5 dspslice (Share 10000)
- Users offset 250 dspslice (1.25 seconds)

```
RACF,TCPIP                         users
||                             ||||||||||
---|---|---|---|---|---|---|---|---|---|---|---|---> (time)
TOD
```

Example 2:   Change tcpip/racf share to ABSOLUTE 20

- TCPIP offset 5 dspslice
- Users offset 84 dspslice (.42 seconds)

```
RACF,TCPIP    users
 ||             ||||||||||
---|---|---|---|---|---|---|---|---|---|---|---|---> (time)

TOD
```

Sample Deadlines - Comparison

Did it make a difference to RACF/TCPIP to reduce share?

- NO. Still number one always on dispatch list

Did it make a difference to users?

- Yes, they are guaranteed 3 times the amount of CPU when looping users are on the system

Does setting shares too high for some users impact other users?

- Only when large CPU consumers (including loopers) exist.
- IBM does not let looping users on their benchmark systems.

Recommend low ABS shares when appropriate for servers

Interactive Bias

SET SRM IABIAS pct nn

- Impacts Traditional workloads only

Improves deadline of first nn dispatch time slices.

- Default of 90 2 gives 90% boost on first time slice, 45% boost on 2nd dispatch time slice.
- Bias range is based on normalized share of highest current dispatchable user
  - If TCPIP is 10% share (scheduled at 10 time slices)
  - user is 1%, (scheduled at 100 time slices)
  - Moves user from 100 time slices delay to 18 time slice delay

Use to improve performance of very interactive CMS users

DOES NOT IMPROVE QUICKDSP Users.

Interactive Bias

Default IABIAS 90 2

- (RACF, tcpip rel share 10000, 10 users rel 100)
- (RACF, tcpip offset 21000/10000 -> 10.5ms)
- (user offset 21000/100 -> 1050 ms)

- 1st time slice offset = offset - (90% * delta) = 115 ms
- 2nd time slice offset = offset - (45% * delta) = 478ms
- 3rd time slice offset = offset = 1050 ms

RACF,TCPIP **2nd** users

|| || **(1st)** |||| ||||||||||

---|---|---|---|---|---|---|---|---|---|---|---|---> (time)

TOD

Delta = difference of best deadline and offset

Analyzing Scheduler/Dispatcher

```
Report: ESASUM       System   z/VM   ESAMAP 4.1.1 01/16/1
Monitor initialized: 03/12/09 at st record analyzed: 03/12/09 05:01:00
--------------------------------------------------------------------------------
Variable Average Minimum Maximum Description
-------- ------- ------- ------- -----------------------------------------------
****************************SCHEDULER PARAMETERS************************
SRMBIASI      90                 Interactive bias intensity percent (SET SRM IAB
SRMBIASD       2                 Interactive bias duration (SET SRM IAB)
SRMTSLIC    5.00                 Minor time slice (ms) (SET SRM DSPSLICE)
SRMTSHOT    2.00                 Minor time slice (ms) for HOTSHOT users
SRMRSCTM  599.90  580.80  659.99 Reset interval (seconds)
SRMABSDL    52.0    48.0    55.0 Total absolute shares of VMDBKs in the dispatch
SRMRELDL     818     550    1900 Total relative shares of VMDBKs in the dispatch

SRMCDLDG       0       0       0 Loading users in dispatch list
SRMLDGUS       5                 Q1 page reads identifying loading user
SRMLDGCP       8                 Loading user capacity of system
SRMP1LDG     100                 Q1 loading user buffer percent (SET SRM LDUBUF)
SRMP2LDG      75                 Q2 loading user buffer percent (SET SRM LDUBUF)
SRMP3LDG      60                 Q3 loading user buffer percent (SET SRM LDUBUF)

SRMP1WSS     300                 Percent memory for E1/E2/E3 users (SET SRM STOR)
SRMP2WSS     300                 Percent memory for E2/E3 users (SET SRM STORBUF)
SRMP3WSS     300                 Percent memory for E3 users (SET SRM STORBUF)
SRMWSSMP    9998                 Maximum working set size percent (SET SRM MAXWSSIZ)

SRMXPCTG       0                 Percent Xstore used in SET SRM STORBUF calculation
SRML1DSP   32767                 Q1/Q2/Q3 Dispatch list size (SET SRM DSPBUF)
SRML2DSP   32767                 Q2/Q3 Dispatch list size (SET SRM DSPBUF)
SRML3DSP   32767                 Q3 Dispatch list size (SET SRM DSPBUF)

SRMEPNF1    2.00    2.00    2.00 E1 expansion factor
SRMEPNF2    2.00    2.00    2.00 E2 expansion factor
SRMEPNF3    2.00    2.00    2.00 E3 expansion factor
SRMLLCNT       0       0       0 Adds per minute to limit list
SRMCONLL       0       0       0 Count of users on limit list
```

ESAMON SHARE MACRO

```
/* calculate normalized share for user */
parse upper arg userid .

ADDRESS ESAMON ’EXTRACT FROM INTERVAL’,
’FIELD RUNTIME NCPUS SYTSCG.SRMRELDL SYTSCG.SRMABSDL MTRSCH.SRMTSLIC’

ADDRESS ESAMON ’EXTRACT USER ’userid,
’FIELD USERDATA.VMDRELSH USERDATA.VMDABSSH’
mtrsch.srmtslic = mtrsch.srmtslic / 4096 / 1000  /* Convert to seconds */
sytscg.srmabsdl = sytscg.srmabsdl * 100 / 64 / 1024 /* Convert from internal
  format */

If SYTSCG.SRMABSDL > 99
  Then factor = 99 / sytscg.srmabsdl  ;   Else factor = 1
If userdata.vmdabssh > 0
 Then normshr = (userdata.vmdabssh * factor)
 Else Do; /* Absolute shares */
   If sytscg.srmreldl = 0 then sytscg.srmreldl = 100
   availshr = (100 – factor * sytscg.srmabsdl)
   normshr = (userdata.vmdrelsh / sytscg.srmreldl) * availshr
End;
say ’Share:’ normshr’%’
say ’deadline:’ mtrsch.srmtslic / (10 * normshr * ncpus ) ’Seconds’
```

```
ESAMON SHARE BARTON
Share: 1.90199309%
deadline: 0.262882133 Seconds Ready;
```

Example

Calculate normalized share for REL share 1000

- Srmabsdl = 50
- Srmreldl = 20000
- Server share is 1000
- (100 – 50) * (1000 / 20000) = 2.5% (1 slice out of 40)

Calculate normalized share when REL share 100

- Srmabsdl = 5
- Srmreldl = 2000
- Linux share is 100
- (100 – 5) * (100 / 2000) = 4.7% (1 slice out of 21)

- **(100 - SRMABSDL) x (relative share / SRMRELDL)**

User Share Case Study

Installation had set TCPIP share from REL 3000 (default) to ABS 3%.

- Good or bad?

What would this do?

- Relative share and absolute share normalized
- Need to know impact on normalized share

What do we want?

- TCPIP to have sufficient share to meet workload requirement

TCPIP needs how much CPU?

45% of one CPU

during peak 15 minutes

```
Report: ESAUSP2      User Resource Rate Report
Monitor initialized: 02/07/07 at 00:00:05 on 2084 serial
---------------------------------------------------------
         <---CPU time--> <----Main Storage (pages)----->
UserID   <(Percent)> T:V <Resident> Lock <-----WSS----->
/Class   Total  Virt Rat Totl Activ  -ed Totl Activ  Avg
-------- ----- ----- --- ---- ----- ---- ---- -----  ----
13:05:00 188.8 178.4 1.1   2M 1559K 4782   2M 1753K  46K
 ***Key User Analysis ***
TCPIP     8.75  6.40 1.4 2722  2722  202  799   799  799
 ***User Class Analysis***
*Keys     0.36  0.32 1.1  527   527    3  558   558  186
*TheUsrs  4.42  4.18 1.1 141K  141K  339 165K  164K  13K
MPROUTE   0.20  0.19 1.1  319   319    1  315   315  472
---------------------------------------------------------
13:26:00 384.2 107.8 3.6   1M 1153K 4384   1M 1442K  37K
 ***Key User Analysis ***
TCPIP    44.83  6.20 7.2 2412  2412  202  621   621  621
 ***User Class Analysis***
*Keys    31.11  0.21 147  160   160    3  338   338  113
*TheUsrs 113.5  2.08  55  64K 64424  229  66K 66305 4973
DTCVSW1  17.69  0.00 .2M   17    17    0   16    16   24
DTCVSW2  16.02  0.00 .2M   17    17    0   16    16   24
```

Server Requirement Case Study

TCPIP used 45% of a processor at peak

- LPAR has 10 processors
- TCPIP has a requirement of 5% of the system to meet peak requirement
- Is 3% absolute sufficient?

What was 3000 relative in normalized terms?

- Calculate normalizeShare =
  - (RelShare / SRMRELDL) * (100 – SRMABSDL) = ???? (6 IFLs….)
- Check ESASUM, Scheduler section

```
Report: ESASUM       System Summary
------------------------------------------------------------------------------
                                   Minimum     Maximum      Std   Obs
Variable Average Minimum Maximum Date  Time  Date  Time      Dev Count Descript
-------- ------- ------- ------- ----- ----- ----- ----- ------ ----- --------
**************************************************SCHEDULER PARAMETERS********
SRMBIASI      90                                                 1394 Interact
SRMBIASD       2                                                 1394 Interact
SRMTSLIC    5.00                                                 1394 Minor ti

SRMTSHOT    2.00                                                 1394 Minor ti

SRMRSCTM  126.38    6.31  306.49 02/07 13:02 02/07 15:54  75.06  1368 Reset in
SRMABSDL     2.3       0     6.0 02/07 00:03 02/07 12:25   80.7  1370 Total ab
SRMRELDL    5296    1200    7070 02/07 15:53 02/07 24:00    523  1370 Total re
```

Dispatch / Eligible List Classes

There are three normal classes, one special class

- used to differentiate types of work
- Control thrashing based on queue
- Q1,Q2,Q3, and Q0

Each class has an associated Elapsed Time Slice (ETS),

- the amount of time a user may stay in the class

Trivial transactions defined as ending transaction in Q1

- ETS adjusted at every qdrop to maintain q1 levels
- Mostly meaningless unit of time (50ms-16sec)
- Defines queue stay, trivial transaction

Dispatch / Eligible List Classes - ETS

Elapsed time slice = .05 - 16 seconds.

- Varies dynamically,

ETS keeps 85% of INQUEUE users in Q1

- Q1 users: Inqueue < 1 ETS
- Q2 users: Inqueue < 7 ETS
- Q3 users: Inqueue > 7 ETS

Q1 size = (Q2 size / 6 + Q3 size / 48) / (.85 / .15)

ETS

- Does not keep '85% of the transactions trivial!
- is not useful to the performance analyst or for SLA!

Dispatch / Eligible List Class 0

Class 0 (Special case, Not held on E–List)

QUICKDSP: set by installation, ETS is the same as Class 2, dispatch priority reprioritized after 8 Q1ETS (1 Q2ETS)

Lockshot: User is holding a lock and stays in class 0 until lock is released. User is treated as QUICKDSP with regard to eligible list.

Hotshot: User is already in queue and interacts with the terminal. Dispatch time slice is a hotshot time slice. Hotshot bias is 90 or 95%

- (users that issue #CP Q T for example during long transaction

User Queue Analysis

```
Report: ESAUSRQ      User Queue Analysis
---------------------------------------------------------------------
                   <-----------Average Number of Users in Queue--------------------->
UserID    <-------Dispatch List-------> Limit  <-------Eligible List------->
/Class       Q0    Q1    Q2    Q3  Ldng List      E0    E1    E2    E3  Ldng
--------  ----- ----- ----- ----- ----- -----  ----- ----- ----- ----- -----
02/12/07
10:01:00    4.0   7.0   7.0  54.0   3.0     0      .     0     0     0     0
Hi-Freq:    4.0  12.2   7.4  48.2   2.0     0      0     0     0     0     .
 ***User
*TheUsrs    0.9   4.8   1.8   9.2   1.0     0      0     0     0     0     .
AAAInter      0   1.0   1.3   9.2   0.3     0      0     0     0     0     .
AAAIDM      0.0   0.3   0.3   0.4   0.1     0      0     0     0     0     .
AAPortal      0     0     0   2.0   0.1     0      0     0     0     0     .
AARPS         0   0.4   0.3     0     0     0      0     0     0     0     .
AAA_Dev     0.1   5.7   3.7  17.4   0.6     0      0     0     0     0     .
 ***Top Users
AAAAP01E      0     0     0   1.0     0     0      0     0     0     0     .
AAAORA2C      0     0     0   3.0   0.1     0      0     0     0     0     .
AAAORA3T      0   0.1   0.7   1.1     0     0      0     0     0     0     .
AAAAP02E      0     0     0   1.0     0     0      0     0     0     0     .
AAAAP02X      0     0     0   1.0     0     0      0     0     0     0     .
```

System domain sample rate: 1 per user per monitor sample

User domain high-frequency sample rate: 1 per second

User data more accurate

VELOCITY SOFTWARE

Dispatcher

Dispatches System VMDBK first

Dispatches user with lowest dispatch deadline priority

- CP System Work
- CP User work
- Users/Servers

Gives a user one dispatch time slice

- Unit of time virtual machine is dispatched
- SET SRM DSPSLICE
- 1-99ms, Default 5ms

Does not care if user is Q1, Q2, or Q3

Processor Local Dispatch Vector

- One per each local processor
- One additional for master

Dispatchable users picked by dispatcher and put on PLDV

- Requires lock, so multiple users “picked”

Moves to master indicates master only services

```
Report: ESAPLDV      Processor Local Dispatch Vector Activity      Velocity Software, Inc.
---------------------------------------------------------------------------------------------------
         <----Users----->   Tran       <VMDBK Moves/sec>  <--------PLDV Lengths-------> Dispatcher
Time     Logged Actv In Q   /sec  CPU  Steals  To Master    Avg  Max Mstr MstrMax %Empty Long Paths
-------- ------ ---- ----  -----   -   ------  ---------   ----  --- ---- ------- ------ ----------
13:16:04    788  274 23.7   19.0   0    126.7      334.3    0.8  2.0  0.3     1.0   44.4      977.4
                                   1     69.5          0    0.1  2.0    .       .   92.5      357.8
                                   2     64.7          0    0.1  2.0    .       .   91.9      315.4
                                   3     69.9          0    0.1  2.0    .       .   91.1      340.6
                                   4     63.2          0    0.1  2.0    .       .   93.5      302.8
                                   5     74.5          0    0.1  2.0    .       .   91.6      383.3
                                       ------  ---------   ----  --- ---- ------- ------ ----------
System:                                 468.5      334.3    1.4 12.0  0.3     1.0  504.9     2677.2
```

E-List to D-List Promotion

To be moved from the Eligible List to the Dispatch List, non-QUICKDSP users must pass three tests.

- DSPBUF q1 q2 q3
- STORBUF q1 q2 q3
- LDUBUF q1 q2 q3

Each test requires that the user fit into a “buffer” based on dispatch list

Q1 user must only pass Q1 tests,

Q2 user must only pass Q1 and Q2 tests, etc

Resource is thus reserved for Q1 / Interactive users

E-List to D-List Promotion

Storage (STORBUF):

- avail = pct * (DPA + Xpct * Xstore) – (Sum Inqueue user working sets)
- If the user’s working set is less than “avail” pages, the test is passed
- “pct” is the q1/q2/q3 percent. Q3 user must pass all three tests.

Paging:

- pct * LDUcapacity < LDUBUF
- If the user is not currently a Loading User, this test is bypassed.
- “pct” is the q1/q2/q3 percent. Q3 user must pass all three tests.

CPU:

- Inq users < DSPBUF
- Default is 32k

E-List to D-List Important Question!

What are we trying to protect with the eligible list?

What does over commitment of Processor mean?

What does over commitment of storage mean?

What does over commitment of paging subsystem mean

z/VM 6.3 changes everything, NEVER want an Elist

SRM Parameters: DSPBUF

SET SRM DSPBUF n1 n2 n3

- Controls number of users in dispatch list
- Defaults to 32k 32k 32k
- Function disabled by definition
- Use SET SHARE instead!

Parameters based on sum of inqueue users:

- 1) Q1+Q2+Q3
- 2) Q2+Q3
- 3) Q3

SRM Parameters: LDUBUF

SET SRM LDUBUF 100 75 60

User defined as Loading if

- n Page faults in one dispatch time slice where “n” is 5 at default
- User logs on (until profile exec completes or one dispatch time slice is used)

SET SRM DSPSLICE changes this algorithm

- Default is 5ms dispatch time slice
- If modify to 1 ms, then “Loading” definition becomes 1 page fault
- If modify to 10ms, then 10 page faults define a loading user

Linux installations have more page devices per server

Loading Capacity

Loading capacity defined as:

- Number of paging devices

SET SRM LDUBUF 100 75 60

- “100” for all users limits number of loading users to number of page devices
- “75” limits number of Q2+Q3 loading users
- “60” for “queue 3” limits number of Q3 loading users to 60% of page devices

PRE 6.3 SRM Parameters: LDUBUF

SET SRM LDUBUF ld1 ld2 ld3

- Controls paging capacity in use by inqueue users
- Defaults to 100 75 60
- (IBM) Recommends disable this function
  - SET SRM LDUBUF 300 200 200
  - **http://www2.marist.edu/htbin/wlvtype?LINUX-VM.30359**
  - SET SRM LDUBUF 100 100 100
- Velocity Software Recommends really using this function (pre 6.3)
  - SET SRM LDUBUF 60 50 40

Death Spiral Example

```
ESAUSRQ      User Queue Analysis                 ESAMAP V2.1.0
----------------------------------------------------------------------------------------
                           <-------------Average Number of Users in Queue------------->
UserID   <--Users-->  Tran <-------Dispatch List-------> <-------Eligible List------->
 /Class  Logged Actv  /min    Q0    Q1    Q2    Q3  Ldng     E0    E1    E2    E3  Ldng
-------- ------ ----  ---- ----- ----- ----- ----- -----  ----- ----- ----- ----- -----
System:    4394  835  3510  17.2  82.3  27.0   4.9   9.3      .  87.6  10.4  78.4    47
Hi-Freq                     13.8  68.8  22.0   5.1   0.4      0  94.9  10.0  67.1     .
NETWORK                      3.0   0.0   0.0   0.7   0.1      0   0.0   0.0  0.00     .
SERVERS                      4.5   0.9   0.2   0.1     0      0   0.2   0.1     0     .
OVMAIL                       6.3  67.8  21.7   4.3   0.3      0  94.6  10.1  0.14     .
**************************************************User Summary**************************
```

Eligible users off the scale, indicates problems

Loading users off the scale, indicates thrashing

And 9 loading users let in, probably a mistake.

So what did they do wrong?

Death Spiral Example

```
DASD Subsystem Analysis Report: ESADSD2      DASD Performance Analysis              Monitor -
-----------------------------------------------------------------------------------------------
 Dev         Device Total  ERP  %Dev <SSCH/sec-> <-------Response times (ms)-------> Qlengths
 No. Serial Type    SSCH SSCH  Busy   avg  peak  Resp  Serv  Pend  Disc  Conn Queue  avg max
---- ------ ------ ----- ----  ---- ----- ----- ----- ----  ----  ----  ----  ---- ----
0E00 VMPG21 3390-3  100K    3  93.2  34.7  46.9  27.1  26.8   0.5  17.9   8.4   0.3  0.0   0
0E01 VMSYS1 3390-3   25K    1  13.9   8.6  16.6  16.2  16.2   0.6  13.1   2.5   0.0  0.0   0
0E03 VMBX01 3390-3  7162    0   2.1   2.5   7.0   8.6   8.6   0.5   4.7   3.4   0.0  0.0   0
0E04 VMEMP1 3390-3   20K    0   9.6   6.9  17.5  15.1  14.0   0.5  10.1   3.4   1.1  0.0   1
0E09 VMPG22 3390-3  100K    3  93.5  34.9  46.4  26.8  26.8   0.5  17.8   8.5   0.0  0.0   0
0E0A VMSPL2 3390-3   52K    0   7.5  18.2  61.8   4.1   4.1   0.6   1.6   2.0   0.0  0.0   0
0E0E VMUS06 3390-3   11K    0   4.3   3.7   7.2  11.9  11.7   0.5   8.1   3.0   0.2  0.0   1
0E10 VMPG26 3390-3  100K    3  92.9  34.7  48.5  26.8  26.8   0.5  17.7   8.6   0.0  0.0   0
0E11 VMPG27 3390-3  100K    3  93.2  34.6  46.9  26.9  26.9   0.5  17.8   8.7   0.0  0.0   0
0E12 VMPG28 3390-3  100K    3  93.2  34.8  49.0  26.8  26.8   0.5  17.8   8.5   0.0  0.0   0
0E14 VMPG23 3390-3  101K    3  93.1  34.9  48.7  26.6  26.6   0.5  17.7   8.4   0.0  0.0   0
0E1A VMUS07 3390-3   13K    0   5.1   4.5  12.6  11.2  11.2   0.5   7.6   3.0   0.0  0.0   0
0E40 VMPG24 3390-3  112K    5  92.7  38.7  52.9  23.9  23.9   0.4  14.8   8.6   0.0  0.0   0
0E42 VMSPL3 3390-3   51K    0   6.6  17.7  38.6   3.7   3.7   0.4   1.4   1.9   0.0  0.0   0
0E48 VMPG25 3390-3  111K    4  92.1  38.7  53.1  23.8  23.8   0.4  14.8   8.6   0.0  0.0   0
0E50 VMPG29 3390-3  111K    4  92.3  38.4  53.3  24.0  24.0   0.4  14.9   8.7   0.0  0.0   0
0E53 VMMTRI 3390-3   37K    1  10.7  13.0  22.2   8.3   8.3   0.4   3.2   4.7   0.0  0.0   0b
```

DASD Paging devices maxed out – All 9 of them

- Nine (9) LOADING USERS CONSUMED
- Nine (9) Paging device capacity!

Should this installation RAISE OR LOWER LDUBUF????

SRM Parameters: STORBUF

SET SRM STORBUF w1 w2 w3

- Controls amount of storage in use by inqueue users
- Defaults:
  - z/VM 5.1: 125 105 95
  - z/VM 6.3:   300 250 200
  - Limits “overcommit” ratio
- Recommendation is always to disable this function,
  - Set to 300 300 300 or similar to match target “overcommit” ratio

SET SRM XSTORE

- Adds this percent of expanded storage to the storage size for purposes of STORBUF calculations. Irrelevant now (6.3)

Thrashing

- Lower LDUBUF

Eligible list and NOT thrashing

- Might be a good thing
- If loading users low, raise (disable) STORBUF
- If loading users high, when DSPSLICE set to 1, raise LDUBUF
  - Check page device utilization

SRM Tuning – Define thrashing?

Thrashing

- Pages per second per user very high
- Page rate consuming DASD

```
Report: ESASSUM      Subsystem Activity
Monitor initialized:                      on 2064 serial 60589     First record
-------------------------------------------------------------------------------
         <---Users----> Transactions <Processor> Storage (MB) <-Paging--> <----
         <-avg number->     Per  Avg. Utilization Fixed Active <pages/sec> <-DAS
Time       On Actv In Q Minute  Resp Total Virt.  User Resid. XStore  DASD Rate
-------- ---- ---- ---- ------ ----- ----- -----  ----- ------ ------ ----- ----
02:03:00   90   64 52.0   69.1 0.224   273   225  40.5 2888.8  14084  9439 2099
02:04:00   91   65 51.0   89.1 0.216   280   230  40.5 2891.9  11781 11702 2621
02:05:00   90   68 49.0   76.0 0.578   276   233  40.5 2893.6  13669  8689 2389
02:06:00   90   62 49.0  107.0 0.145   277   232  41.6 2895.3  13050  9333 2268
02:07:00   90   64 49.0  119.0 0.259   282   232  40.6 2892.1  12717 10610 2451
02:08:00   90   62 50.0  103.0 0.228   278   227  40.5 2891.6  11932 11412 2595
02:09:00   90   62 50.0  106.9 0.249   277   227  40.5 2891.0   6699 13035 3143
02:10:00   90   64 50.0  113.1 0.157   267   207  40.8 2885.1   7957 14753 3656
02:11:00   90   63 48.0   66.0  9.88   239   174  41.0 2875.5   5006 13913 3390
**************************************************Summary********** ***********
Average:   90   64 49.8   94.4 0.996   272   221  40.7 2889.4  10765 11432 2735
```

SRM Tuning – Define thrashing?

Thrashing

- CPU wait and Page wait are UN-Productive
- Users in CPU wait may have pages stolen

```
Report: ESAXACT      Transaction Delay Analysis
-------------------------------------------------------------------------
                     <---------------Percent non-dormant-----------------
UserID   <-Samples->                      E-  D-  T-     Tst <Asynch>
/Class   Total  In Q Run Sim CPU SIO Pag SVM SVM SVM  CF Idl I/O Pag Ldg
-------- ----- ----- --- --- --- --- --- --- --- --- --- --- --- --- ---
12/22/03
02:03:00    90     52 5.8   0  44   0  33   0   0 1.9 5.8 3.8 5.8       .
Hi-Freq:  5369  3410 4.8 0.5  40 0.4  29   0  13 0.9   0 5.9 3.5 0.4 2.0
 ***Key User Analysis ***
TCPIP       59     58 1.7  14 5.2   0   0   0   0   0   0   0  79   0   0
 ***User Class Analysis***
Comms      236    115 0.9 7.0 3.5   0 0.9   0   0   0   0 0.9  79   0 7.8
FILESRVR   236    236  16 0.8  43   0  24   0   0   0   0  10 0.4 1.3 4.7
LNXDBASE   236    236 4.2 0.4  56   0  30   0   0   0   0 8.1 0.4   0 0.4
LNXWEB    1298   1298 4.2 0.3  54   0  37   0   0   0   0 2.4 0.9   0 0.7
SFS        177      1   0   0   0   0   0   0   0   0   0   0   0   0   0
SYSMON     177     60 1.7   0 3.3  23 1.7   0   0   0   0  70   0   0   0
```

SRM Parameters: SET QUICKDSP

SET QUICKDSP userid ON | OFF

- “ON” means by pass the eligible list
- Disables scheduler’s ability to manage resource abusers
- Does NOT impact user’s share

IBM sometimes recommending turning QUICKDSP on for all Linux servers – disabling the scheduler that allows hundreds or thousands of users to work well together….

- Puts all servers into Q0

The scheduler controls the death spiral, disabling this control is unhealthy

Only use for servers that are “shared resource”

- TCPIP, RACF servers for example
- Database servers serving many users or other servers

SRM Parameters: Linux Peculiarities

Prior to Timer Pop patch:

- Linux pops 100 times per second
- **Always** in queue 3
- Working set never trimmed so presumed very large
- Storage buffer must account for ALL servers
- SET SRM STORBUF High for queue 3

After timer pop patch

- Linux pops 1 time per second
- Idle servers should be in queue 1
- Idle servers in queue < 50% of time
- Working set trimmed
- Storage buffer requirement smaller
- SET SRM STORBUF not an inhibitor, default may suffice

z/VM 6.3 makes this irrelevant from storage management perspective

Other QDrop Issues

Virtual Multi-processors:

- Both virtual processors must go idle for server to drop from queue
- Analysis required.

Current JDK polls every 10ms

Current polling issues impact:

- WAS/Java,
- DOMINO,
- Tivoli Applications

Storage management changes in 6.3 make polling less relevant

Understanding Scheduler

Enable Scheduler domain for user

Record Raw Monitor data for analysis interval

Run ESAMAP against raw data

Set ESAMAP Option:

- TRACE.USER = ’userid’

ESATUNA LISTING

- QDrops
- QAdds
- Transaction Details
- Seek Details

Transaction Time Line

When analyzing a performance problem, build a timeline

A CMS “short” transaction timeline

```
07:11:00.459272 Scheduler Data (SCLAEL), Add User to Eligible List: 1
07:11:00.459436 Scheduler Data (SCLADL), Add User to Dispatch List: 1
Dispatch lists: q0: 1 q1: 1 q2: 0 q3: 1
07:11:00.461404 Scheduler Data (SCLRDC), Read Complete From 0004
07:11:00.464087 Scheduler Data (SCLWRR), Write Response To 0004
07:11:01.924552 Scheduler Data (SCLDDL), Drop User from Dispatch List
```

1. Add user to Eligible List (SCLAEL)
2. Move user to dispatch list SCLADL)
3. Read input data from screen (SCLRDC)
4. Write input data back to screen (SCLWRR)
5. Drop user from dispatch list (SCLDDL)

Tracing User

ESATUNA Report

Very large

Time stamped

Details of activity

(Transactions cut at beginning of next transaction)

```
07:10:00.878347 Sample Data (USEACT), Resources used:
07:10:00.878506 Sample Data (USEINT), Delay Analysis
07:10:08.842449 Event Data (USETRE) response times:
Response time (seconds): 1.827
InQueue time (seconds): 2.224
Think time (seconds): 27.5
07:10:08.842501 Event Data (USEATE), Resources used:
07:10:08.842584 Event Data (USEITE), Wait Analysis:
07:11:00.459018 Event Data (USETRE) response times:
Response time (seconds): 0.122
InQueue time (seconds): 2.018
Think time (seconds): 49.6
07:11:00.459067 Event Data (USEATE), Resources used:
User operating in ESA mode.
User has Relative Share of: 100
Processor Consumption (CPU Seconds)
TotCPUTm 0.02020 VirtCPU 0.00269
Storage Consumption (Pages)
PagesRes 235.000 WSS Size 235.000 VM Size 2048.00
Paging Activity (Counts)
NonPfPgs 43.0000
Spooling Activity (Counts)
SplPages 55.0000
Non-DASD Virtual I/O (Counts)
Cons I/O 2.00000
07:11:00.459144 Event Data (USEITE), Wait Analysis:
InQueue State Sample Counts
InQueue 2.00000 TstIdle 2.00000
InQueue Percent State Analysis
InQueue 3.84615 TstIdle 100.000
Queue Analysis
Pct Q1 100.000
Time slice used up in Q1 1 times.
```

Tracing Linux User

```
17:57:45.583123 VCPUad: 00 Scheduler Data (SCLAEL), Add User to Eligible List: 1
17:57:45.583126 VCPUad: 00 Scheduler Data (SCLADL), Add User to Dispatch List: 1
 Dispatch lists: q0:   4 q1:   5 q2:   0 q3:  27
 Dispatch Priority(Original): 2833969.0000
  Dispatch Priority(Revised): 2833967.0000
 Elapsed time slice:       0.4658 Required thruput:     422.0000
  VMDIABIA: Interactive Bias in effect
17:57:45.773364 VCPUad: 01 Scheduler Data (SCLAEL), Add User to Eligible List: 1
17:57:45.773367 VCPUad: 01 Scheduler Data (SCLADL), Add User to Dispatch List: 1
 Dispatch lists: q0:   4 q1:   6 q2:   2 q3:  27
 Dispatch Priority(Original): 2833969.0000
  Dispatch Priority(Revised): 2833967.0000
 Elapsed time slice: 1799808.0000 Required thruput:     455.0000
  VMDIABIA: Interactive Bias in effect
17:57:45.773416 VCPUad: 01 Scheduler Data (SCLDDL), Drop User from Dispatch List
 User requires scheduler intervention, VMDSACTL = 00001000
  VMDIDROP: Drop from DISP Immediately
  VMDIABIA: Interactive Bias in effect
17:57:46.048896 VCPUad: 00 Scheduler Data (SCLDDL), Drop User from Dispatch List
 User requires scheduler intervention, VMDSACTL = 00000001
  VMDRSCEL: VMDBK exceeded limits of controlled resource
User requires scheduler intervention, VMDSACTX = 00010000
  VMDESEND: Elapsed Timeslice Exceeded
  VMDIABIA: Interactive Bias in effect
```