

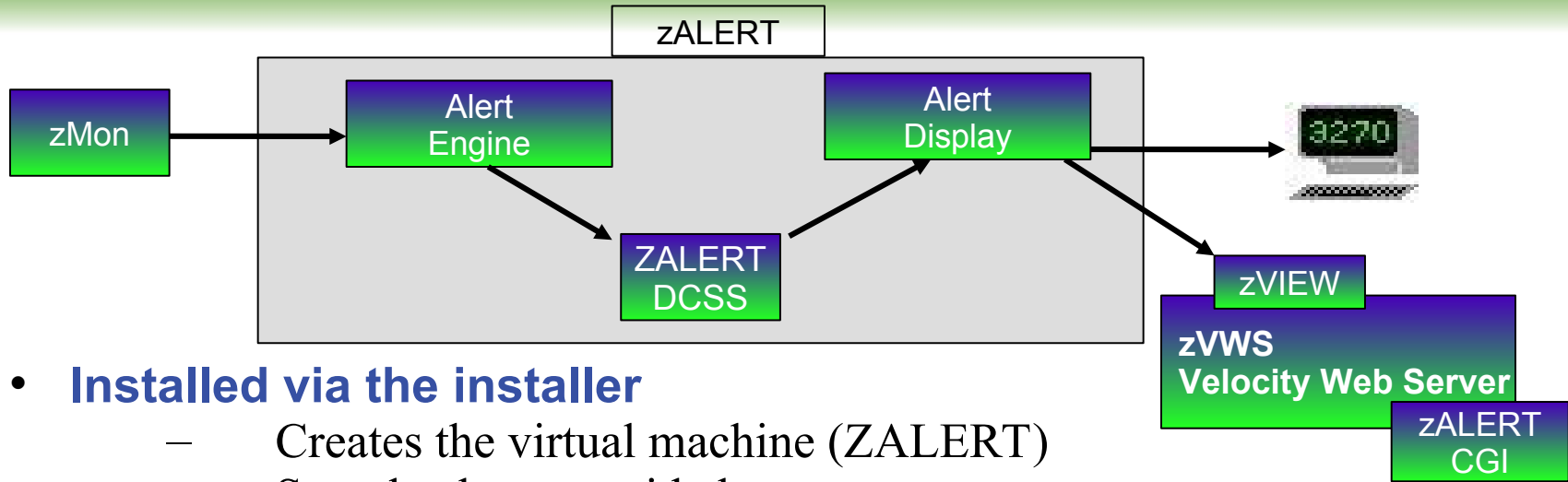
zVPS Alerts

Richard Smrcina
Velocity Software, Inc.
Performance Workshop
June, 2025

Agenda

- **Overview**
- **What are alerts?**
 - ◆ Where do alerts fit
- **zAlert technology**
- **Alert samples**
- **Defining your own alert**
 - ◆ CPU Utilization
 - ◆ LPAR Utilization
- **Notification**
 - ◆ MSG to Operator
 - ◆ SNMP trap
- **Advanced topics**

- **A separately installed component of zVPS**
 - Included as part of the product suite
- **An alert is an indication of an abnormal condition**
- **An abnormal condition can be**
 - Exceeding a certain threshold
 - An object in an incorrect state
 - Volume offline
 - Virtual machine not logged on
 - Incorrect system settings



- **Installed via the installer**
 - Creates the virtual machine (ZALERT)
 - Sample alerts provided
 - More samples on the web site
- **Requires a DCSS for operation**
- **Alert messages stored in the DCSS**
- **Message retrieval handled by a separate EXEC**

- **The alert virtual machine**
 - Wakes up every minute, processes alerts
 - Generates alert messages
 - Sends out required notifications

- **Each of the defined extracts is executed**
 - Values returned from extracts is compared against user defined thresholds
 - Message displayed and action taken when thresholds are exceeded

- Terminal session

– ZALERT [alertfile]

```
vmlink .dir vmsysvps:zmon.code  
ZALERT
```

```
Screen: RKS2LV RKS2LV 12 Jun 2017 10:31:38  
----- Exceptions Analysis Alerts -----  
Type Description  
APSP Page space is 16.72% used  
DSCK Node linux001 fs '/usr' is at 87%  
DSCK Node linux002 fs '/' has 120244K available  
DSCK Node linux002 fs '/var' is at 89% with 5528K avail  
ESAD ESAMON DCSS utilization is 2.6%  
LNCP CPU utilization on Linux node lxora12 is 5.14%  
LNDX /usr area on linux001 is 86.65% full  
LNDX /var area on linux001 is 80.60% full  
LNDX /usr area on linux002 is 86.65% full  
LNDX /var area on linux002 is 88.58% full  
SPOL Spool utilization is 60% (above 10)  
VMC2 User ZALERT used 0.1761 CPU sec (0.2935%)  
XACP Processor utilization at 1.0%
```

PF7=Backward PF8=Forward PF3=Quit F5=Prev Alert F6=Next Alert
3279 01/001

Viewing alerts - ZVIEW

- **zView**

- Select 'zAlert Definitions'
- Select alert file to display



zALERT Definitions

- VSIVM4
- LINALERT
- VMALERT

VSIVM4 - Exceptions Analysis Alerts - 18/06/19 at 09:56 - DEMO

Code	Alert Description
CHEK	Spool Utilization is 7%
ESAD	ESAMON DCSS utilization is 33.5%
JHPU	JVM 'AppSrv01-server1' on lxoral2 Heap Utilization 29.8%
JHPU	JVM 'AppSrv01-server1' on lxoral2b Heap Utilization 27.6%
LNCPU	CPU util on Linux node suselnx2 is 47%
LNDX	Filesystem / on REDHAT6X is 95% full
LNDX	Filesystem /root/r73/repo on TESTRL74 is 100% full
LNPR	CPU Utilization for process smallstr-3702 on suselnx2 is 42%
LNSU	Swap util for Linux node lxoral2 is 100%
LNSU	Swap util for Linux node lxoral2b is 100%
LNSU	Swap util for Linux node oracle is 45%
LNSU	Swap util for Linux node sles12 is 100%
LPCCP	LPAR VSIVM4 CPU Utilization is 115%
LPCCP	LPAR VSIVM5 CPU Utilization is 67%
LXDN	Node s11s2ora is down
LXDN	Node ZSXL0150 is down
ORPG	DB orcl on oracle PGA Utilization 43%
ORPG	DB db01 on sles12 PGA Utilization 70%
ORPG	DB db02 on sles12 PGA Utilization 73%
ORSW	DB db02 on sles12 System IO Waits 1 Time 0.000
VMCP	User SUSELNX2 CPU Utilization is 47.3%
VMCW	User ZALERT is in 100% CPU wait
VMLP	User SLES12 may be looping; CPU 18%, loop count 595
VMSW	User ZTCP is in 25% simulation wait
XACP	Processor utilization is 114.7%

Viewing alerts - CGI

- CGI placed in the ZVWS.ROOT directory

`http://<vm-host>/zalert.cgi`

`http://<vm-host>/zalert.cgi?file=<alertfile>`

VELOCITY SOFTWARE

Exception Analysis Alerts

Velocity Software - VSIVM4

Alert File: VSIVM4
System: VSIVM4
Select: VSIVM4

19 Jun
09:30:59

Type	Description
CHEK	Spool Utilization is 7%
ESAD	ESAMON DCSS utilization is 33.5%
JHPU	JVM 'AppSrv01-server1' on Ixora12 Heap Utilization 27.6%
JHPU	JVM 'AppSrv01-server1' on Ixora12b Heap Utilization 27.8%
LNCP	CPU util on Linux node mail is 46%
LNDX	Filesystem / on REDHAT6X is 95% full
LNDX	Filesystem /root/r73/repo on TESTRL74 is 100% full
LNSU	Swap util for Linux node Ixora12 is 100%
LNSU	Swap util for Linux node Ixora12b is 100%
LNSU	Swap util for Linux node oracle is 45%
LNSU	Swap util for Linux node sles12 is 100%
LPCP	LPAR VSIVM4 CPU Utilization is 72%
LPCP	LPAR VSIVM5 CPU Utilization is 66%
LXDN	Node s11s2ora is down
LXDN	Node ZSXL0150 is down
ORPG	DB orcl on oracle PGA Utilization 43%
ORPG	DB db01 on sles12 PGA Utilization 70%
ORPG	DB db02 on sles12 PGA Utilization 73%
VMLP	User SLES12 may be looping; CPU 20%, loop count 569
VMPG	Page rate for user ZWEB05 69
XACP	Processor utilization is 71.6%



- **Alert samples are shipped with the ZALERT package**
 - ◆ ALERT1 MONALERT is the primary sample file
 - ◆ Older sample files are shipped with the filetype MONSAMP
 - VMALERT, LINALERT, HEALTH and HEALTH2
 - ◆ Samples check various conditions that can potentially occur
 - CPU/Spool/Page Utilization, I/O Rate, Paging Rate
 - Node CPU utilization, I/O Rate, Disk utilization, Swap rate and utilization
- **Additional samples available on our web site**

Defining your own alerts

- **Coding an alert requires the use of data fields maintained by zVPS**
- **Data is extracted from the monitor**
- **Analyzed to determine if it exceeds a threshold**
- **For values greater than threshold**
 - ◆ Message issued
 - ◆ Optional action is taken
- **Alerts generally use the following statements**
 - ◆ EXTRACT
 - ◆ VAR
 - ◆ ALERT
 - ◆ LEVEL
 - ◆ TEXT

- **Alert for CPU Utilization**

Extract

```
Parms CPU TOTAL
```

'Extract' is the beginning of an alert definition or set of alert definitions

```
var  cpu_serial | 6    | serial
var  util       | 5 1  | sytprp.cpuutil
```

```
alert util xacp
level 00  green
level 20  yellow
level 40  pink
level 80  red
text Processor utilization at &util%
```

- **Alert for CPU Utilization**

Extract

Parms CPU TOTAL

PARMS determines the type of data to extract

```
var  cpu_serial | 6      | serial
var  util       | 5 1    | sytprp.cpuutil
```

```
alert util xacp
level 00 green
level 20 yellow
level 40 pink
level 80 red
text Processor utilization at &util%
```

- **Alert for CPU Utilization**

Extract

Parms CPU TOTAL

```
var  cpu_serial | 6   | serial
var  util       | 5 1 | sytprp.cpuutil
```

```
alert util xacp
level 00  green
level 20  yellow
level 40  pink
level 80  red
```

```
text Processor utilization at &util%
```

Fields to extract -
names are described in the PDR
(Performance Data Reference)

Can be a single field or multiple
fields involved in simple to
complex math operations.

- **Alert for CPU Utilization**

```
Extract  
Parms CPU TOTAL
```

Variables defined for use
in the following alerts

```
var  cpu_serial  | 6 | serial  
var  util        | 5 1 | sytprp.cpuutil
```

Size of each variable with
optional decimal precision

```
alert util xacp  
level 00 green  
level 20 yellow  
level 40 pink  
level 80 red  
text Processor utilization at &util%
```

- Alert for CPU Utilization

```
Extract
```

```
Parms (ALERT statement defines  
a specific alert
```

```
var cpu_serial | 6 | serial  
var util | 5 1 | sytprp.cpuutil
```

```
alert util xacp
```

Four character code used when displaying alerts

```
level 00 green
```

```
level 20 yellow
```

```
level 40 pin  
level 80 red
```

Each alert requires a previously defined variable

```
text Processor utilization at &util%
```

- Alert for CPU Utilization

Extract

Param

LEVEL statement controls the threshold values

```
var cpu_serial | 0 | serial
var util | 5 1 | sytprp.cpuutil
```

```
alert util xacp
```

Color of the alert text when this level is exceeded

```
level 00 green
```

```
level 20 yellow
```

Values tested against the alert variable

```
level 40 pin
```

```
level 80 red
```

```
text Processor utilization at &util%
```

- **Alert for CPU Utilization**

Extract

Parms CPU TOTAL

```
var  cpu_serial | 6 | serial
var  util       | 5 1 | sytprp.cpuutil
```

```
alert util xacp
level 00 green
level 20 yellow
level 40 pink
level 80 red
```

```
text Processor utilization at &util%
```

Message displayed on
3270 and zView alert screens

Alert variable
substitution

- The 3270 screen based on the alert definition

The screenshot shows a 3270 terminal window titled 'x3270-2 RKS2LV'. The window has a menu bar with 'File' and 'Options'. The main display area shows the following text:

```
Screen: RKS2LV 15 Jun 2015 07:48:51
-----
Type Description
XACP Processor utilization at 2.3%
```

Annotations with callout boxes:

- 'Alert file being displayed' points to the 'Screen:' field.
- 'Code specified on ALERT statement' points to 'XACP'.
- 'TEXT directive with variable substitution' points to 'Processor utilization at 2.3%'.

At the bottom of the screen, there is a status bar with the text: PF7=Backward PF8=Forward PF3=Quit. In the bottom right corner, there is a small icon and the text ':00.2 001/001'.

- **Adjust the number and value of levels based on local requirements**
 - ◆ At least one LEVEL statement is necessary
 - ◆ LEVEL statements are evaluated bottom to top
- **Standard 3270 colors are allowed**
 - ◆ Turquoise, Blue, Red, Yellow, Green, Pink, White
 - ◆ If no color is specified, the default is Green
 - ◆ Color modifiers are allowed
 - **REV**video – reverse video
 - **BL**ink – blink the entire text
 - **UNDERLINE** – underline the entire text

LPAR Utilization

- Alert for LPAR Utilization

ESALPARS - Logical Partition Analysis Summary - VSIVM4																		
<Complex>		-----Logical Partition-----					-----Assigned Shares-----											
Time	Phy Dspth	CPU Slice	Name	Virt Nbr	CPUs Typ	<%Assigned> Total	<---LPAR---> Ovhd	<VCPU Pct> Weight	<VCPU Pct> /SYS	<VCPU Pct> /CPU	Cap- ped	Abs Cap	Wait Comp	<--Thread--> Idle	Entitld CPU Cnt			
08:38:00	7	Dynam	Totals:	4	CP	18.3	0.1	125	100						2			
08:38:00	7	Dynam		13	IFL	305.5	1.6	665	100						4			
08:38:00	7	Dynam	VSIVM4	04	3	IFL	193.1	1.3	200	30.08	10.03	40.1	No	No	No	123.27	2	1.20
08:38:00	7	Dynam	VSIVC1	07	1	IFL	96.8	0.0	20	3.01	3.01	12.0	No	No	No	1.45	2	0.12
08:38:00	7	Dynam	VSIVC2	08	1	IFL	0.7	0.0	100	15.04	15.04	60.2	No	No	No	0.61	2	0.60

Extract

Parms LPAR *

Criteria sytcup.lcupname <> 'Totals:'

var lpname | 8 | sytcup.lcupname

var lputil | 3 0 | sytcup.pctcpu

alert lputil lpcp

level 70 yellow

level 85 red

level 95 red rev

text LPAR utilization of &lpname is &lputil%

- Alert for LPAR Utilization

```
Extract
```

```
Parms LPAR *
```

```
Criteria sytcup.lcupname <> 'Totals:'
```

```
var lpname | 8 | sytcup.lcupname
```

```
var lputil | 3.0 | sytcup.lcupname
```

Data filtering

```
alert lputil lpcp
```

```
level 70 yellow
```

```
level 85 red
```

```
level 95 red rev
```

```
text LPAR utilization of &lpname is &lputil%
```

- Alert for LPAR Utilization

```
Extract
```

```
Parms LPAR *
```

```
Criteria sytcup.lcupname <> 'Totals:'
```

```
var lpname | 8 | sytcup.lcupname
```

```
var lputil | 3 0 | sytcup.pctcpu
```

```
alert lputil lpcp
```

```
level 70 yellow
```

```
level 85 red
```

```
level 95 red rev
```

```
text LPAR utilization of &lpname is &lputil%
```

Text will be in reverse video
(black text, red background)

- Alert for LPAR Utilization display

3270

```
Screen: ALERTLPR          Velocity Software - VSIVM5          4 Jan 2017 11:36:12
----- Exceptions Analysis Alerts -----
Type Description
LPCP LPAR VSIVM4 CPU Utilization is 78%
```

zView

ALERTLPR - Exceptions Analysis Alerts - 17/01/04 at 11:36 - VM5

Code	Alert Description
LPCP	LPAR VSIVM4 CPU Utilization is 78%

Defining your own alert – Complex operations

- **Numerous fields can be combined using math operations**
 - Statements can be continued with a dash

```
extract
parms user *
criteria userdata.userid <> 'System:' & useact.vmdtttime > 0
var    userid      | 8    | userdata.userid
var    cpuutil     | 3 1 | useact.vmdtttime * 100 / RUNTIME
var    io_rate     | 6 0 | (useact.vmdvdsct+useact.vmdvosct-
      +useact.vmdvcsct+useact.vmdvusct-
      +useact.vmdvtsct)/runtime
var    page_rate   | 6    | (useact.vmdctpgr+useact.vmdctpgw)/RUNTIME
var    exp_store   | 8    | useact.vmdctxrd+useact.vmdctxwt
var    userprt     | 8    | useact.vmdctpgr
var    vmdtttime   | 5 2 | useact.vmdtttime
```

Defining your own alert - Second vdisk usage

- Using two swap disks with different priority
 - ◆ Second disk larger than the first
 - ◆ First disk fills, Linux uses the second disk
 - ◆ Alert when second disk is used

ESAVDSK - VDISK Analysis - RKS2LV															
Time	Owner	Space Name	<--Size-->		<--pages-->		Prv	VIO	<AddSpce>			<--DASD-->			
			AddSpc	VDSK	Resi-	Lock-			or	rate	Usr	Cre-	Del-	Sto-	pages/s
			Pages	Blks	dent	ed	Shr	/min	Lks	ates	etes	len	Read	Write	
07:56:00	LINUX001	VDISK\$LINUX001\$0202\$0031	4000	32000	407	0	Shr	311	1	0	0	41.2	48.0	38.6	
07:56:00	LINUX001	VDISK\$LINUX001\$0203\$0032	16000	128K	8093	0	Shr	845	1	0	0	37.6	172.5	36.6	
07:56:00	LINUX002	VDISK\$LINUX002\$0202\$0053	4000	32000	0	0	Shr	0	1	0	0	0	0	0	

Vdisk activity indicator



Defining your own alert - Second vdisk usage

- **Create an alert to show Vdisk activity**

- ◆ Only care about the second disk

```
extract
parms space vdisk* user *
criteria stoasi.mdiovdev = '0203'
var   userid   | 8   | aspace.userid
var   vdev     | 4   | stoasi.mdiovdev
var   io_rate  | 6   | stoasi.qdiocnt
```

Select address spaces beginning with vdisk

Common second virtual disk

```
alert io_rate lsvd
level 0 red
text Node &userid is using the second virtual disk
```

Screen: **LSVD**

RKS2LV

----- Exceptions Analysis Alerts -----

Type Description

LSVD Node LINUX001 is using the second virtual disk



- **A notification is a message sent to interested parties of an alert condition**
- **Sent in one or more of the following forms**
 - ◆ CP MSG/MSGNOH
 - ◆ Email
 - ◆ Text page (via email)
 - ◆ SNMP Trap

- **At it's simplest a notification can take the form of a message to a CMS user**

```
alert userprt vmpg | count &userid  
level 5 green action CP MSG OP &code &text  
text Page rate for &userid is &userprt/sec (above &tlevel for &tcount)
```

ACTION keyword on the LEVEL statement allows targeted messaging for a specific threshold

```
09:27:10 ZALERT VMPG Page rate for OPERATOR is 6.8/sec (above 5 for 6)
```

- **SNMP Trap configuration**

- ◆ Create/Modify SNMP TRAPDEST on the CONFIG disk

* following is default 1.3.6.1.4.1.15601
192.168.5.182 velocity 2B06010401F971 ;

- ◆ Use the TRAP directive on the LEVEL command

```
alert spool_use spol
level 10 green
level 70 yellow trap &code &atext
level 80 pink
level 90 red
text Spool utilization is &spool_use% (above &tlevel)
```

- SNMP Trap result

```

Screen: RKS2LV                                     RKS2LV                                     6
----- Exceptions Analysis Alerts -----
Type Description
APSP Page space is 26.51% used
DVRT I/O rate for volume VM5W01 0124 103.35/sec
DVRT I/O rate for volume VM5PG1 0127 7.72/sec
ESAD ESAMON DCSS utilization is 3.3%
LNCP CPU utilization on Linux node sles12 is 22.66%
LNDX /usr area on linux001 is 86.65% full
LNDX /usr area on linux002 is 86.65% full
LNDX /var area on linux002 is 88.36% full
LNPU Process stresser CPU usage on node sles12 is 20.65%
LNSU Swap utilization for Linux node sles12 is 24%
PGRT System paging rate 48 (above 5)

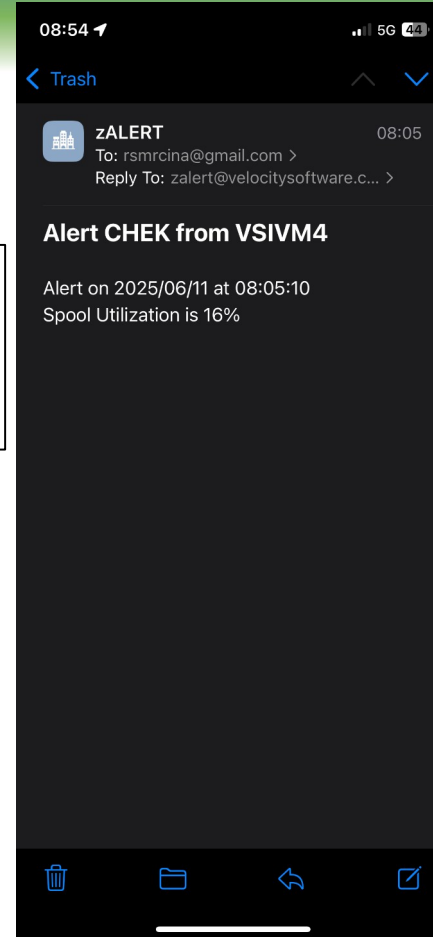
VMCP User ZVPS is at 1.8807%
VMC2 User RKSDEV used 0.0018 CPU sec (0.0030%)
VMC2 User ZALERT used 0.2047 CPU sec (0.3412%)
VMIO I/O rate for user SFSZVPS 17
VMPG Page rate for OPERATOR is 6.9/sec (above 5 for 5)
VMPG Page rate for SMTP is 5.5/sec (above 5 for 1)
VMPG Page rate for ZALERT is 10.6/sec (above 5 for 1)
XACP Processor utilization at 3.1%
    
```

ID	Severity	Time	Node	Interface
217	Normal	Jan 6, 2017 9:41:00 AM		192.168.5.48
uei.opennms.org/generic/traps/EnterpriseDefault Edit notifications for event				
Trap from 192.168.5.48 Type: 0 Message: SPOL Spool utilization is 72% (above 70)				

- **Email**

VSIVM4 MONALERT:

```
alert spool_use chek
limit 359 1 | &cpu_serial
email joeadmin@mydomain.com
level 1 green notify
text Spool Utilization is &spool_use%
```



- **Options add additional function at the alert level**

- Priority
- Log
- Count
- Separated on alert directive with a vertical bar

ALERT DISKPCT LNDX | <options>

- **Log - Writes alert text displays to a file**
- **Count - Tallies the number of times an alert appears on the display**

- **Priority alters the display sequence of an alert or level**
 - Priority is a numeric value 1-9, default is 3

```
extract
parms node *
criteria hstmem.used > 0
var    node    | 8    | tcpip.node
var    memused | 6 2 | (hstmem.used/hstmem.size)*100
var    desc    | 16   | hstmem.desc
function diskpct | 6 2 | &node &memused &desc

alert diskpct lndx | pri 2
level 5  green
level 50 yellow
level 80 pink
level 90 red rev pri 6
ltext &desc area on &node is getting really full!! (&diskpct%)
text &desc area on &node is &diskpct% full
```

Priority

```
x3270-2 RKS2LV
File Options
Screen: ALRT6 RKS2LV 15 Jun 2015 10:58:34
----- Exceptions Analysis Alerts -----
Type Description
LNDX /var area on linux001 is getting really full!! (90.18%)
APSP APSP Page space is 12% used (10)
LNDX / area on linux001 is 37.92% full
LNDX /boot area on linux001 is 19.01% full
LNDX /local area on linux001 is 21.36% full
LNDX /usr area on linux001 is 86.65% full

PF7=Backward PF8=Forward PF3=Quit
:00.1 001/001
```

```

alert io_rate dvrt | log
level 5 green
level 10 blue
level 20 turquoise
level 30 pink
level 40 red rev
text I/O rate for volume &volser &rdev &io rate/sec
    
```

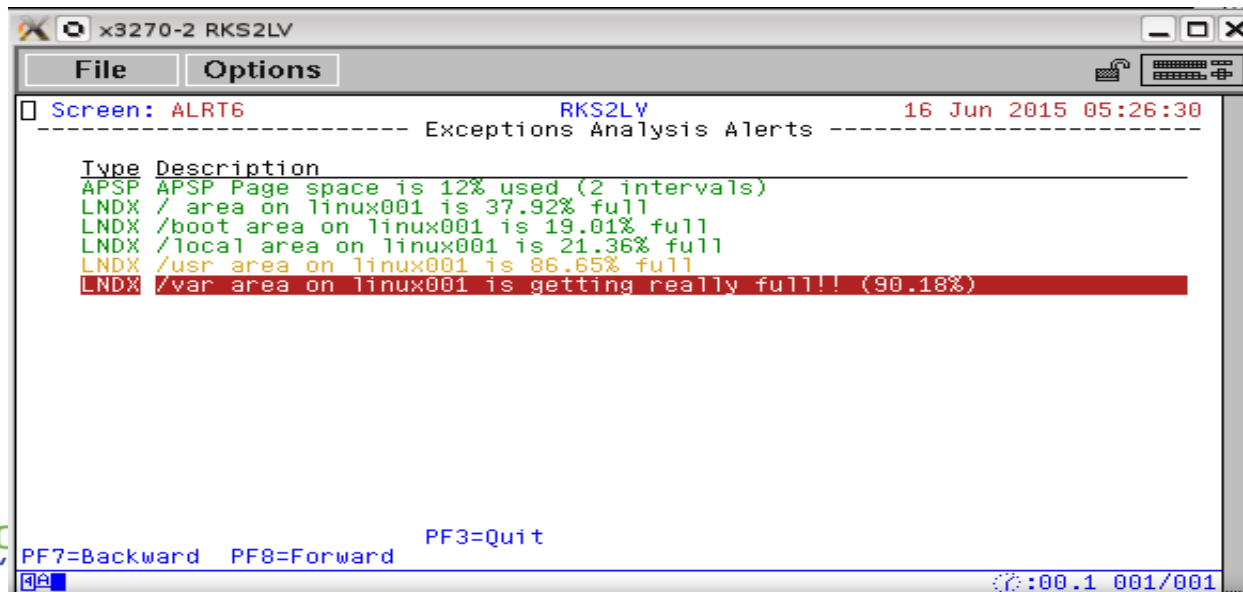
The screenshot shows a terminal window titled 'x3270-2 RKS2LV'. The window contains a log of I/O rates for various volumes. The log header is 'AL150614 LOGDATA E1 V 114 Trunc=114 Size=116 Line=2 Col=1 Alt=0'. The log entries are as follows:

Time	Device	Operation	Volume	Rate
00:00:13	DVRT	I/O rate	VM5W01 0124	7.17/sec
00:00:13	DVRT	I/O rate	VM5WK1 0126	262.43/sec
00:00:13	DVRT	I/O rate	VM5PG1 0127	59.07/sec
00:00:13	DVRT	I/O rate	VM5WK2 0128	43.37/sec
00:01:12	DVRT	I/O rate	VM5W01 0124	131.87/sec
00:01:12	DVRT	I/O rate	VM5W02 0125	6.35/sec
00:01:12	DVRT	I/O rate	VM5WK1 0126	214.42/sec
00:01:12	DVRT	I/O rate	VM5PG1 0127	13.12/sec
00:01:12	DVRT	I/O rate	VM5WK2 0128	271.92/sec
00:02:10	DVRT	I/O rate	VM5W01 0124	35.10/sec
00:02:10	DVRT	I/O rate	VM5WK1 0126	60.60/sec
00:02:10	DVRT	I/O rate	VM5PG1 0127	8.18/sec
00:02:10	DVRT	I/O rate	VM5WK2 0128	219.78/sec
00:04:10	DVRT	I/O rate	VM5PG1 0127	10.82/sec
00:34:11	DVRT	I/O rate	VM5PG1 0127	5.38/sec
01:01:10	DVRT	I/O rate	VM5PG1 0127	7.70/sec
01:04:10	DVRT	I/O rate	VM5PG1 0127	6.48/sec
02:01:10	DVRT	I/O rate	VM5PG1 0127	5.02/sec
02:04:10	DVRT	I/O rate	VM5PG1 0127	5.05/sec

The terminal window also shows a menu bar with 'File' and 'Options' buttons, and a status bar at the bottom with 'X E D I T 1 File' and a timestamp '00:00.3 023/007'.

Count

```
alert page_use apsp | count
level 10 green
level 30 yellow
level 50 red
text &code Page space is &page_use% used (&tcount intervals)
```



The screenshot shows a terminal window titled "x3270-2 RKS2LV" with a menu bar containing "File" and "Options". The terminal content displays "Screen: ALRT6" and "RKS2LV" with a timestamp of "16 Jun 2015 05:26:30". Below this is a header "----- Exceptions Analysis Alerts -----". A table follows with two columns: "Type" and "Description". The table lists several alerts, with the last one highlighted in red: "LNDX /var area on linux001 is getting really full!! (90.18%)". At the bottom of the terminal, there are navigation instructions: "PF7=Backward PF8=Forward PF3=Quit" and a status bar showing ":00.1 001/001".

Type	Description
APSP	APSP Page space is 12% used (2 intervals)
LNDX	/ area on linux001 is 37.92% full
LNDX	/boot area on linux001 is 19.01% full
LNDX	/local area on linux001 is 21.36% full
LNDX	/usr area on linux001 is 86.65% full
LNDX	/var area on linux001 is getting really full!! (90.18%)

- **Disable removes an alert from evaluation**
 - Used in a maintenance situation when unwanted alerts or false alerts may be triggered
- **Enable is the opposite of disable**
 - The default and need not be specified
 - Provided for consistency

```
alert page_use amsp
disable
level 10 green
level 30 yellow
level 50 red
text Page space is &page_use% used
```

- **The LIMIT directive delays an ACTION for the specified number of intervals**

```
extract
var serial      | 6    | system.serial
var spool_use   | 3 0 | (sytasg.calslti2*100)/sytasg.calslta2

alert spool_use spol
limit 4 1 | &serial
level 70  yellow
level 80  red
level 90  red rev ACTION CP MSG ZVPS Spool Util is &spool_use%
text Spool Utilization is &spool_use%
```

- The **LIMIT** directive delays an **ACTION** for the specified number of intervals

```
extract
var serial      | 6      | system.serial
var spool_use   | 3 0 | (sytag.calslti2*100)/sytag.calslta2

alert spool use spo1
limit 4 1 | &serial
level 70     yellow
level 80     red
level 90     red rev ACTION CP MSG ZVPS Spool Util is &spool_use%
text Spool Utilization is &spool use%
```

Key field

Number of intervals
to delay executing ACTION

After the delay, number of
intervals TO execute ACTION
(default is 1)

- **This LIMIT directive:**

```
limit 4 1 | &serial
```

- **Will delay ACTION for 4 intervals**
- **Execute ACTION for 1 intervals**
- **Repeat**
- **For example, when started at 11:55**

```
12:00:29 * MSG FROM ZALERT: SPOOL UTIL IS 95%
```

```
12:05:30 * MSG FROM ZALERT: SPOOL UTIL IS 95%
```

```
12:10:31 * MSG FROM ZALERT: SPOOL UTIL IS 95%
```

First message is
delayed 4 intervals

One interval of
ACTION

- LIMIT initial action setting**

```
extract
var serial      | 6   | system.serial
var spool_use   | 3 0 | (sytag.calslti2*100)/sytag.calslta2

alert spool use spol
limit 4 1 1 | &serial
level 70 yellow
level 80 red
level 90 red rev ACTION CP MSG ZVPS Spool Util is &spool_use%
text Spool Utilization is &spool_use%
```

Number of intervals the
action is taken before
the delay

- **An alert can call an external process**
 - ◆ Function
 - ◆ Stage
- **Function is a REXX EXEC that processes already extracted data**
 - ◆ Called for each record returned from an extract
 - ◆ Returns a single value
- **Stage is an EXEC that is called as a pipeline stage**
 - ◆ Must have a filetype of REXX
 - ◆ Can independently run it's own extract
 - ◆ Returns a single value or plugs the result into defined alert variables

- Function is specified in place of 'var'
- Variable name is an EXEC that is called
- Alert variable is the EXEC name

```
extract
parms node *
criteria hstmem.used > 0
var    node    | 8    | tcpip.node
var    memused | 6 2 | (hstmem.used/hstmem.size)*100
var    desc    | 60   | hstmem.desc
function diskpct | 6 0 | &node &memused &desc

alert diskpct lndx
level 20 green
level 50 yellow
level 80 pink
level 90 red rev
text Filesystem &desc on &node is at &diskpct%
```

- **Function is specified in place of 'var'**

```
extract
parms node *
criteria hstmem.used > 0
var   node   | 8   | tcpip.node
var   memused | 6 2 | (hstmem.used/hstmem.size)*100
var   desc   | 16  | hstmem.desc
function diskpct | 6 0 | &node &memused &desc

alert diskpct lndx
text Filesystem &desc on ct%
level 20 green
level 50 yellow
level 80 pink
level 90 red rev
```

Parameters passed
as exec args

Function definition
is the exec called
and the variable
used in the alert

- **REXX exec called from the alert**

```
/* DISKPCT EXEC: Filter out memory or read-only filesystems */  
parse arg node pct descr .
```

```
firstword = word(descr,1)  
rptzero = 'Real Memory Swap Physical Virtual Cached'
```

```
if wordpos(descr,rptzero) > 0 then  
  pct = 0
```

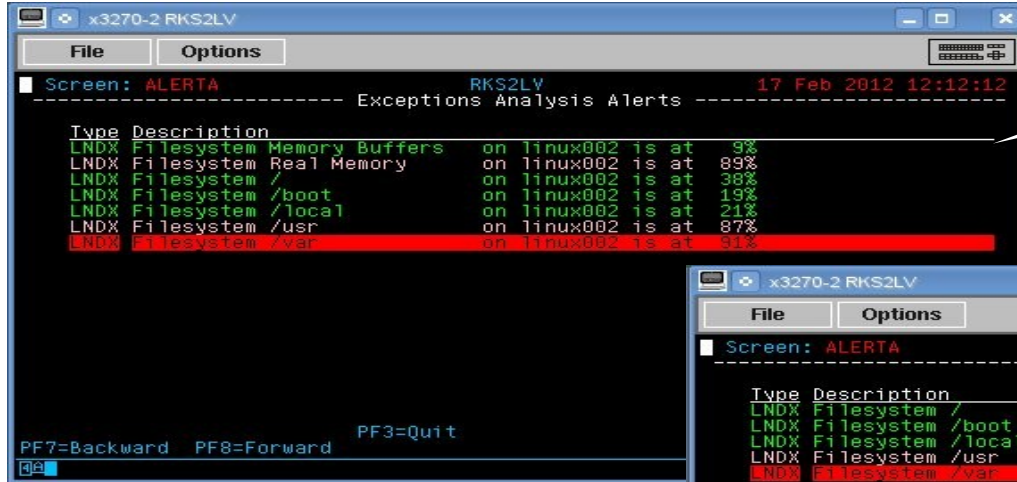
```
if left(descr,6) = '/media' then  
  pct = 0
```

```
return pct
```

Parameters passed
from alert

Value returned
to the alert

- Results of function call



x3270-2 RKS2LV

File Options

Screen: ALERTA RKS2LV 17 Feb 2012 12:12:12

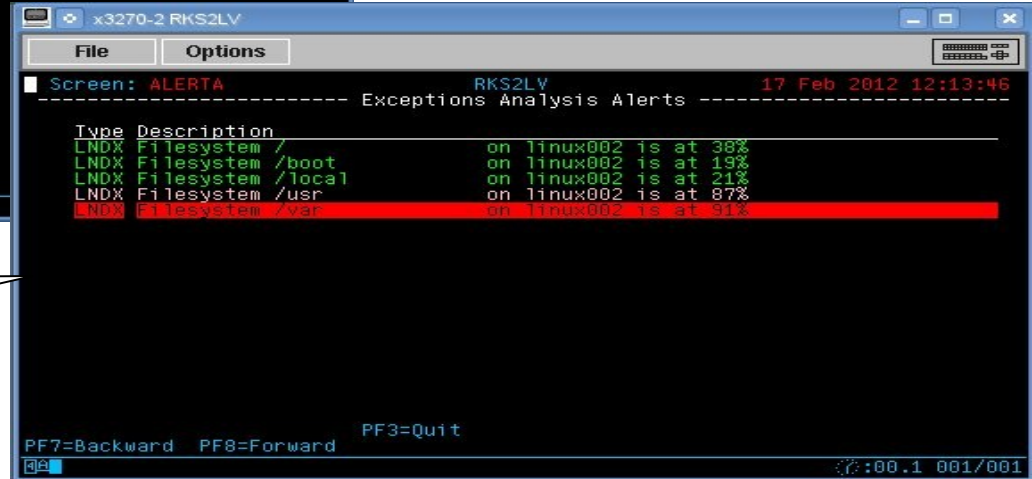
----- Exceptions Analysis Alerts -----

Type	Description	on	linux002	is	at	%
LNDX	Filesystem Memory Buffers	on	linux002	is	at	9%
LNDX	Filesystem Real Memory	on	linux002	is	at	89%
LNDX	Filesystem /	on	linux002	is	at	38%
LNDX	Filesystem /boot	on	linux002	is	at	19%
LNDX	Filesystem /local	on	linux002	is	at	21%
LNDX	Filesystem /usr	on	linux002	is	at	87%
LNDX	Filesystem /var	on	linux002	is	at	91%

PF7=Backward PF8=Forward PF3=Quit

Original display

With DISKPCT EXEC



x3270-2 RKS2LV

File Options

Screen: ALERTA RKS2LV 17 Feb 2012 12:13:46

----- Exceptions Analysis Alerts -----

Type	Description	on	linux002	is	at	%
LNDX	Filesystem /	on	linux002	is	at	38%
LNDX	Filesystem /boot	on	linux002	is	at	19%
LNDX	Filesystem /local	on	linux002	is	at	21%
LNDX	Filesystem /usr	on	linux002	is	at	87%
LNDX	Filesystem /var	on	linux002	is	at	91%

PF7=Backward PF8=Forward PF3=Quit

:00.1 001/001

External Processing

- **Detection mechanism for required virtual machines**
 - ◆ Service machines
 - ◆ Utility machines
 - ◆ Linux systems

```
extract
var    dummy    | 1 | 1
stage alrtmusr | 8 |

alert dummy xmvm
level 0 red action CP MSG OP &code &text
text User &alrtmusr not logged onto system
```

```
Screen: TOP20 RKS2LV
----- Exceptions Analysis Alerts -----
Type Description
XMVM User ZWEB06 not logged onto system
```

MISSING USER

```
/* VELOCITY Virtual Machines
ZSERVE ZTCP
ZADMIN ZWEB01 ZWEB02 ZWEB03
ZWEB04 ZWEB05 ZWEBLOG
ZWEB06
/*
/* SFS service machines
/*
VMSERVU VMSERVS SFSZVPS
```

- Since z/OS data is integrated into the architecture, writing alerts is very easy

```
Screen: ZOS Velocity Software - VSIVM4 17 Jun 2019 06:52:15
----- Exceptions Analysis Alerts -----
```

Type	Description
ZCEC	IFL CPU Utilization 102.0
ZSIO	VSI1 I/O Rate 26.5
ZSYS	VSI1 CPU Utilization 20.2

```
extract
parms zos * CPU TOTAL
criteria zoscpu.samples > 0
var sysid | 8 | zosid.sysid
var cpuutil | 6 1 | zoscpu.pctbusy
var cpuutiln | 6 1 | zoscpu.pctbusy / zoscpu.samples

alert cpuutiln zsys
level 2 blue
level 8 yellow
level 80 red
text &sysid CPU Utilization &cpuutil
```

TCP/IP (customer request)

- Failed attempts

```
extract
parms node *
var   userid      | 8   | tcpip.node
var   fails       | 3 1 | tcp.attemptfails
```

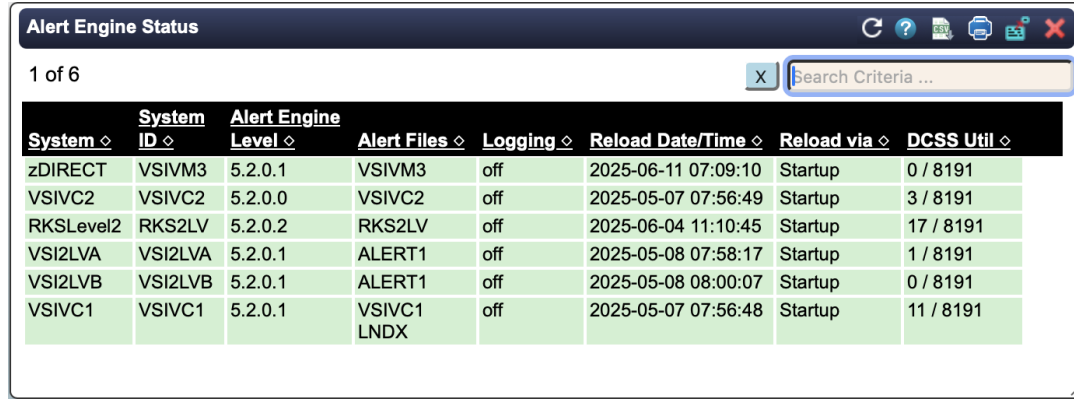
```
alert fails tcpf
level 10 blue
level 20 blue rev
level 40 yellow rev
level 80 red rev
text User &userid TCP fail count &fails
```

TCPF User s11s2ora TCP fail count 11.9

ATTMPTFAILS	flt	=	flt 0	TCP Connections to CLOSED from SYNSENT/ SYN-RCVD
-------------	-----	---	-------	--

- **Many ZALERT functions can be controlled via SMSG**
 - CMS
 - CP
 - QUERY
 - REREAD
 - RESTART
 - SET
 - STATUS
 - STOP

- **Updates in 5200+**
 - Global LOG option (customer request)
 - Logs all alerts to the spool
 - Add ‘End of Display’ to SMSG results (customer request)
 - Use zPRO common email routine
 - Updates to stale data processing (customer request)
 - More integration with zPORTAL



The screenshot shows a window titled "Alert Engine Status" with a search bar and a table of system configurations. The table has 8 columns: System ID, System Level, Alert Files, Logging, Reload Date/Time, Reload via, and DCSS Util. The table contains 7 rows of data.

System ID	System Level	Alert Files	Logging	Reload Date/Time	Reload via	DCSS Util	
zDIRECT	VSIVM3	5.2.0.1	VSIVM3	off	2025-06-11 07:09:10	Startup	0 / 8191
VSIVC2	VSIVC2	5.2.0.0	VSIVC2	off	2025-05-07 07:56:49	Startup	3 / 8191
RKSLevel2	RKS2LV	5.2.0.2	RKS2LV	off	2025-06-04 11:10:45	Startup	17 / 8191
VSI2LVA	VSI2LVA	5.2.0.1	ALERT1	off	2025-05-08 07:58:17	Startup	1 / 8191
VSI2LVB	VSI2LVB	5.2.0.1	ALERT1	off	2025-05-08 08:00:07	Startup	0 / 8191
VSIVC1	VSIVC1	5.2.0.1	VSIVC1 LNDX	off	2025-05-07 07:56:48	Startup	11 / 8191

- **Updates in 5100**
 - Expanded wildcard support
 - Similar to PCRE (for include/exclude)
 - Value difference alert
 - Calculates the difference from current value and previous value
 - Configurable stale count



Rich Smrcina
Velocity Software, Inc
rich@velocitysoftware.com