zVPS Alerts

Richard Smrcina
Velocity Software, Inc.
Performance Workshop
June, 2019

VELOCITY SOFTWARE

PROVEN PERFORMANCE

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**

What are alerts?

- **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

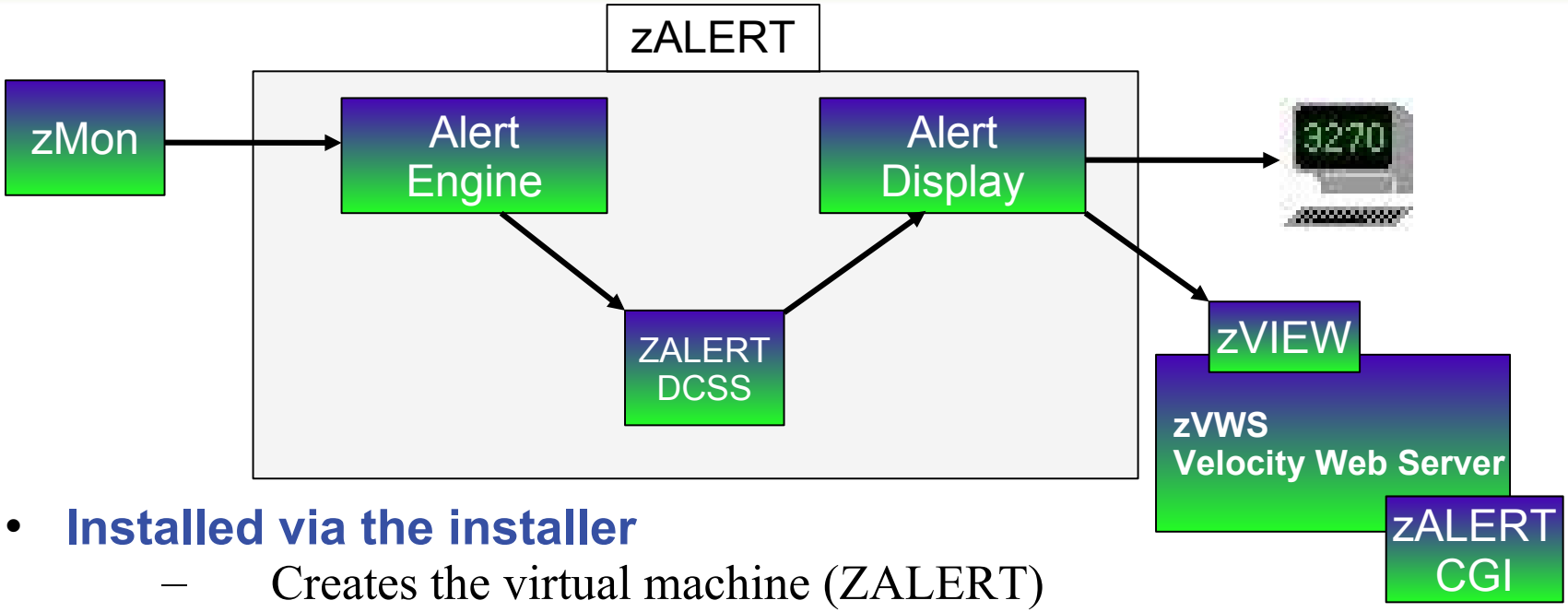
Overview

This presentation goes through the finer points of alert processing.

Where alerts come from, how they are used and specified in the product.

Alerts are no good if they need to be visually watched or monitored... notifications provide a proactive mechanism to using alerts.

zALERT Technology



- **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**
- **zAlert 4.1 (MONALERT) can still be used as is, but is functionally stabilized**

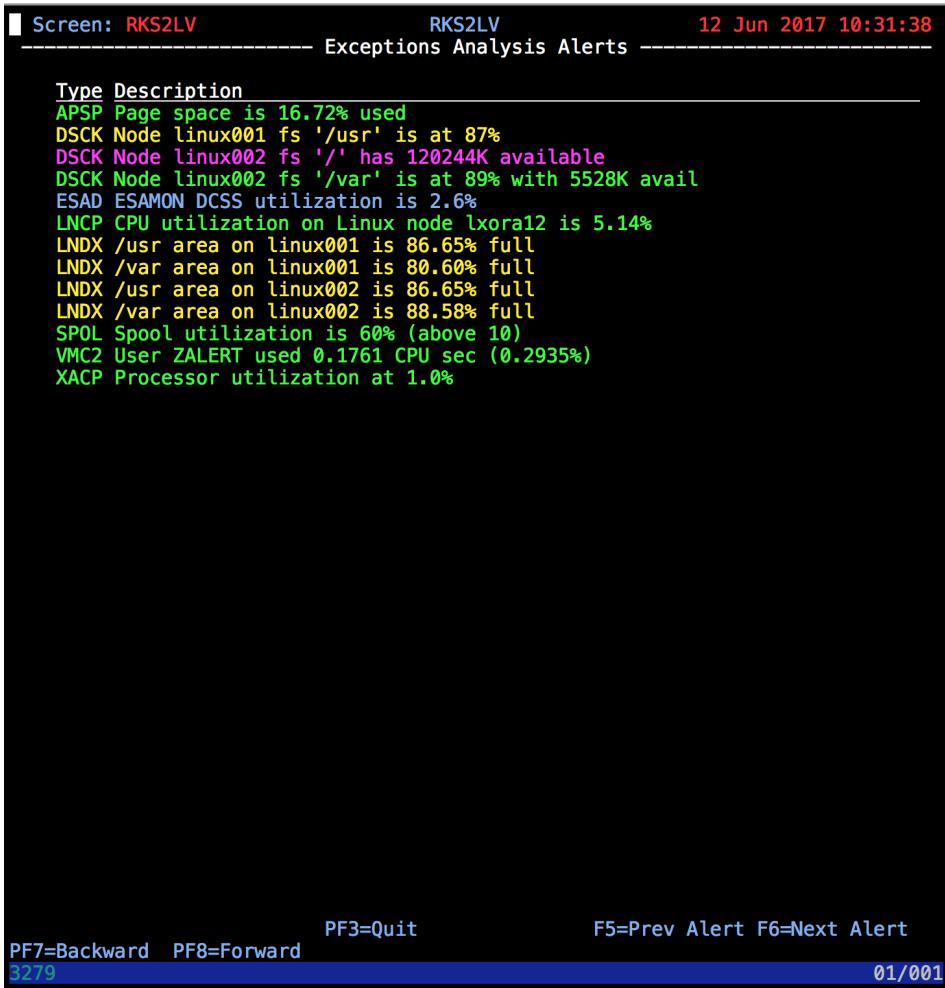
Alert processing

- **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

Viewing alerts - 3270

- **Terminal session**
  - ZALERT [alertfile]

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



Viewing alerts - ZVIEW

- **zView**
  - Select 'zAlert Definitions' →
  - Select alert file to display →

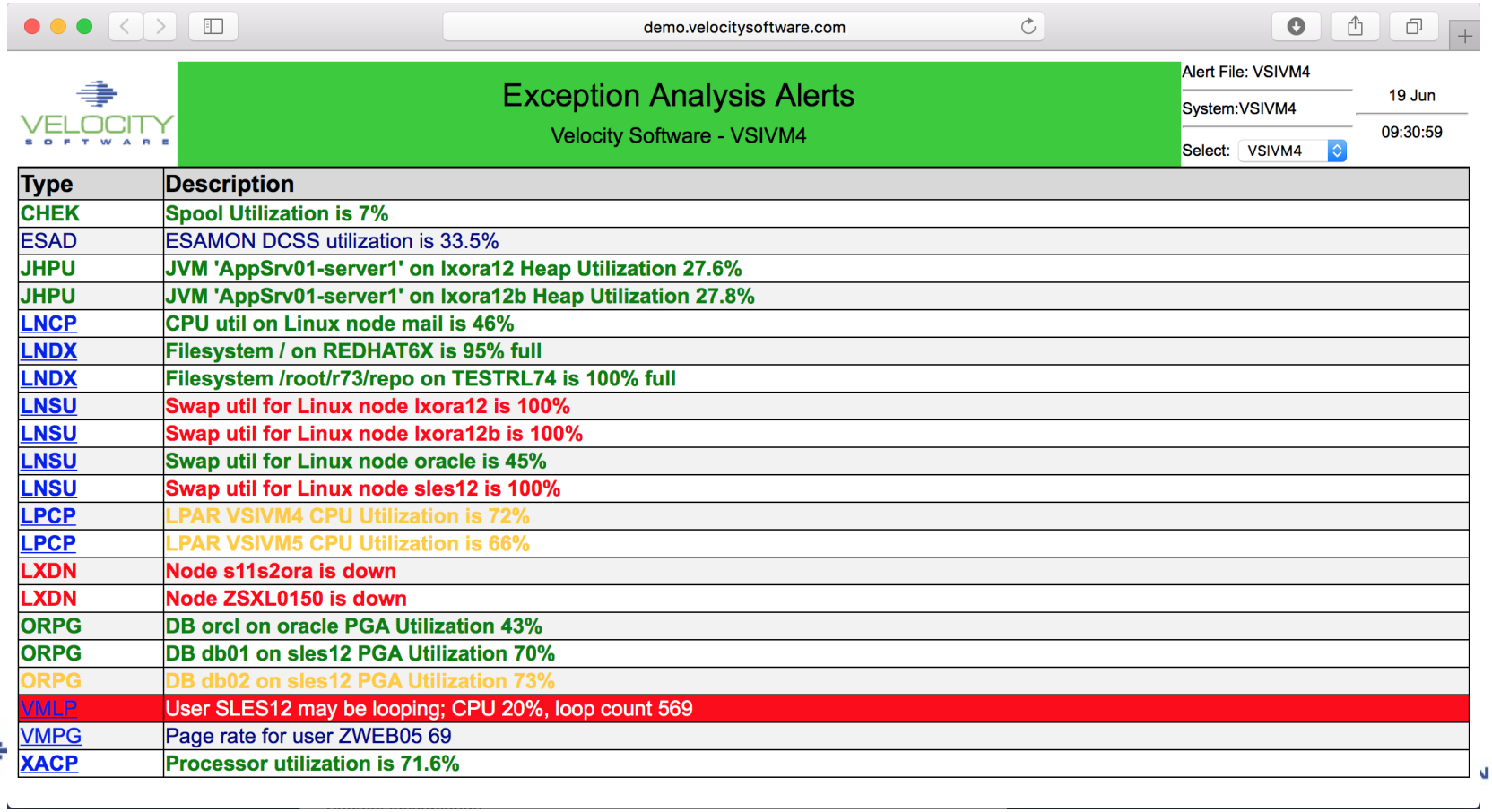
VSIVM4 - Exceptions Analysis Alerts - 18/06/19 at 09:56 - DEMO

Code	Alert Description
CHEK	Spool Utilization is 7%
ESAD	ESAMON DCSS utilization is 33.5%
JHPU	JVM 'AppSrv01-server1' on lxora12 Heap Utilization 29.8%
JHPU	JVM 'AppSrv01-server1' on lxora12b Heap Utilization 27.6%
LNCP	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 lxora12 is 100%
LNSU	Swap util for Linux node lxora12b 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 115%
LPCP	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>
```



Type	Description
CHEK	Spool Utilization is 7%
ESAD	ESAMON DCSS utilization is 33.5%
JHPU	JVM 'AppSrv01-server1' on lxora12 Heap Utilization 27.6%
JHPU	JVM 'AppSrv01-server1' on lxora12b 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 lxora12 is 100%
LNSU	Swap util for Linux node lxora12b 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

- **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

Defining your own alerts

- **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%
```

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

Defining your own alerts

- **Alert for CPU Utilization**

```
Extract
Parms CPU TOTAL

var  cpu_serial | 6 [illegible] | 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%
```

PARMS determines the type of data to extract

Defining your own alerts

- **Alert for CPU Utilization**

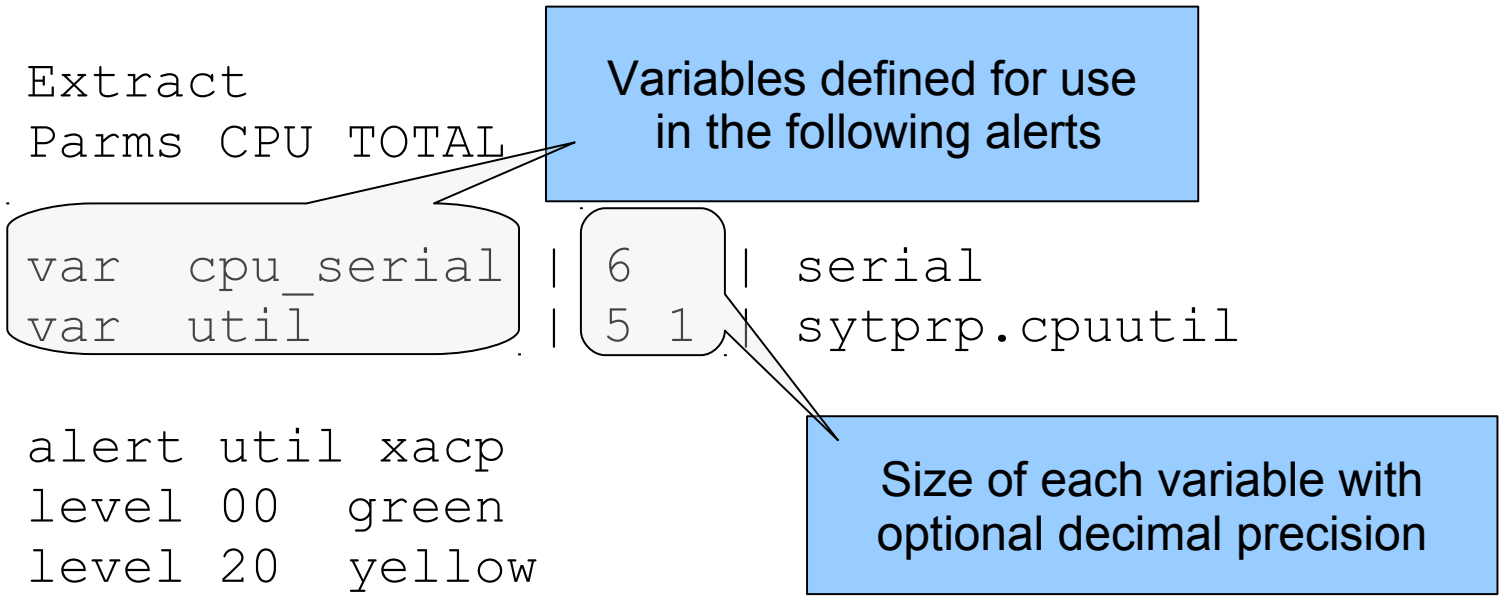
```
Extract
Parms CPU TOTAL

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
```

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.

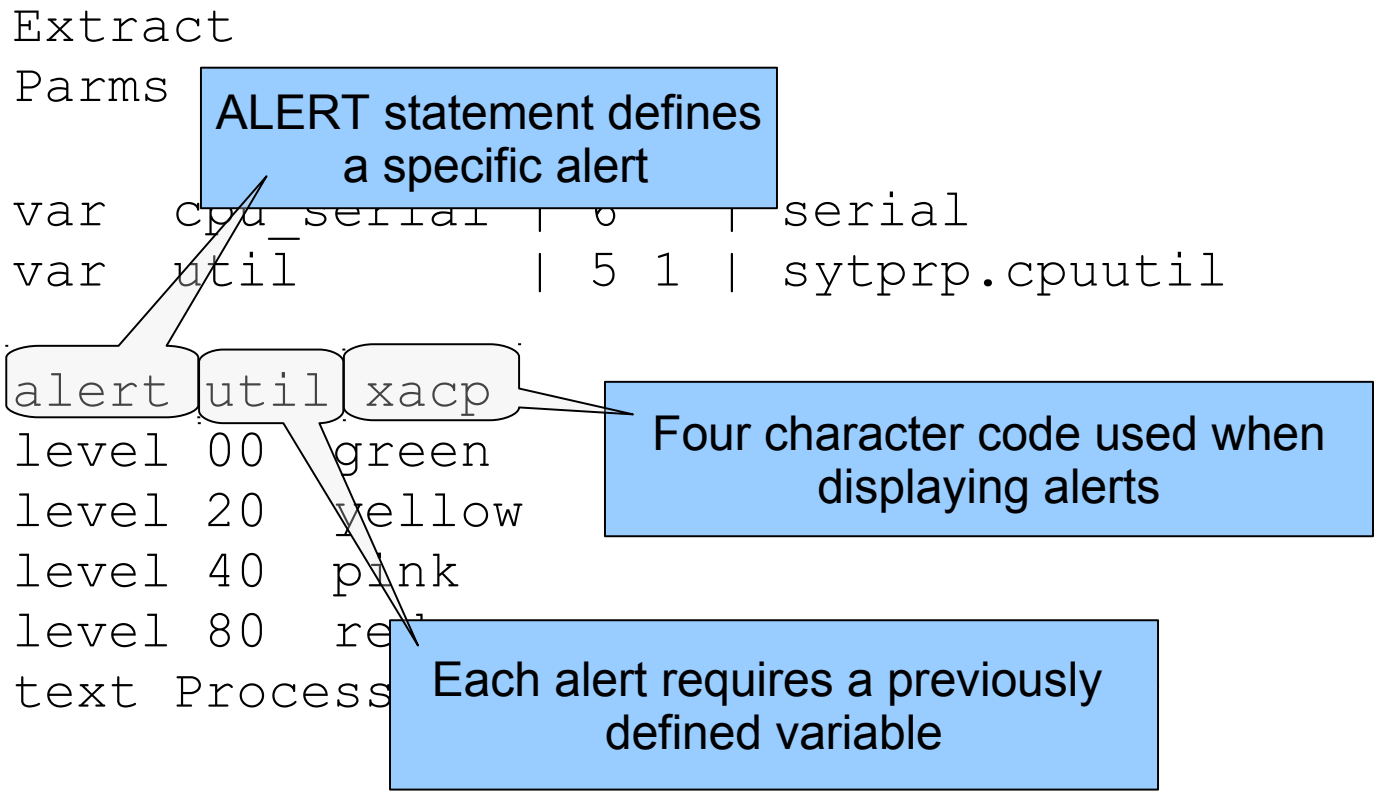
Defining your own alerts

- **Alert for CPU Utilization**



Defining your own alerts

- **Alert for CPU Utilization**



Defining your own alerts

- **Alert for CPU Utilization**

```
Extract
Parms CPU TOTAL

var                                  serial
var  util          | 5 1 |  sytprp.cpuutil

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

LEVEL statement controls the threshold values

Color of the alert text when this level is exceeded

Values tested against the alert variable

Defining your own alerts

- **Alert for CPU Utilization**

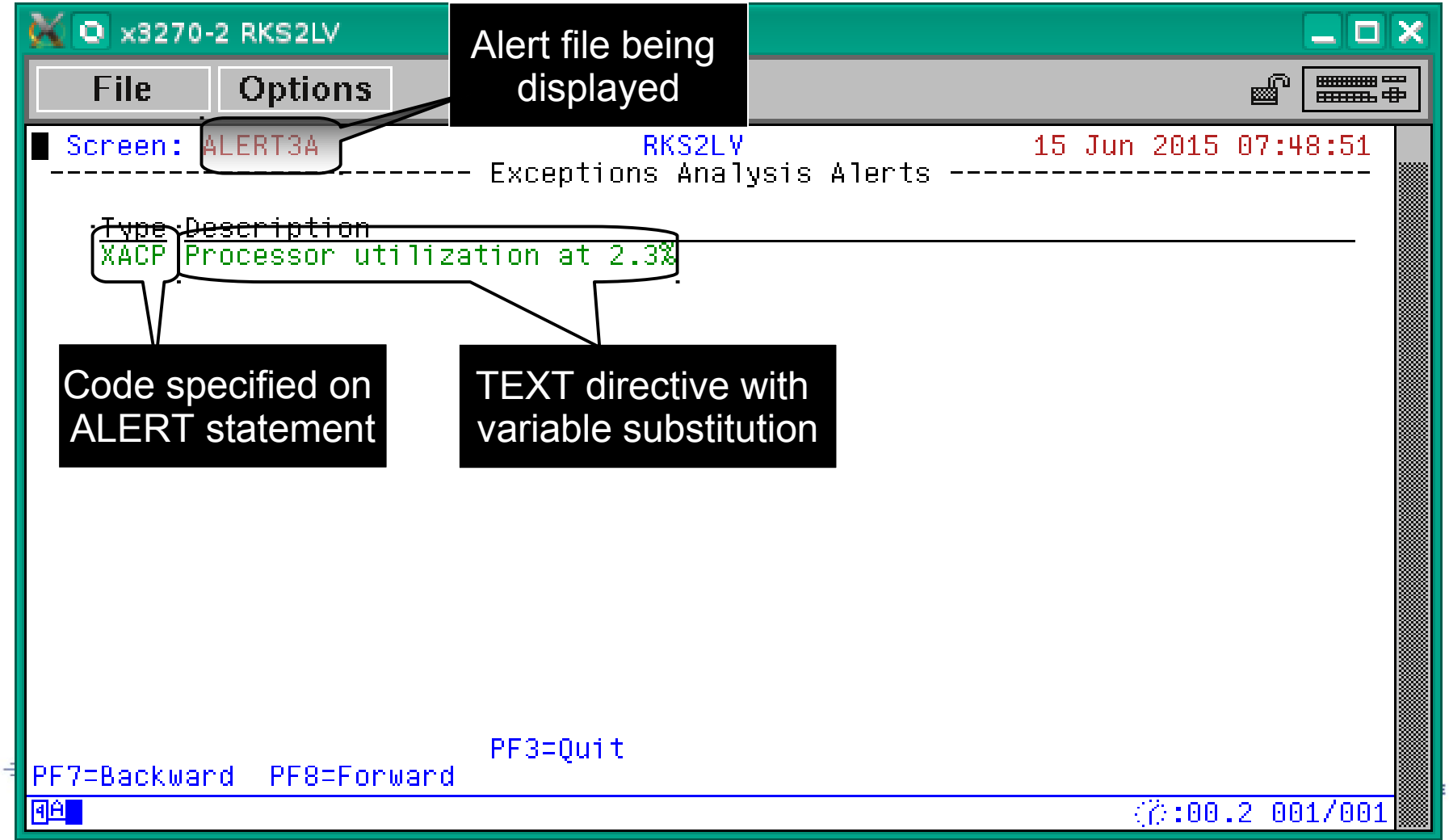
```
Extract
Parms CPU TOTAL

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

Alert result

- **The 3270 screen based on the alert definition**



Alert rules

- **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
    - **BLI**nk – blink the entire text
    - **UNDERLINE** – underline the entire text

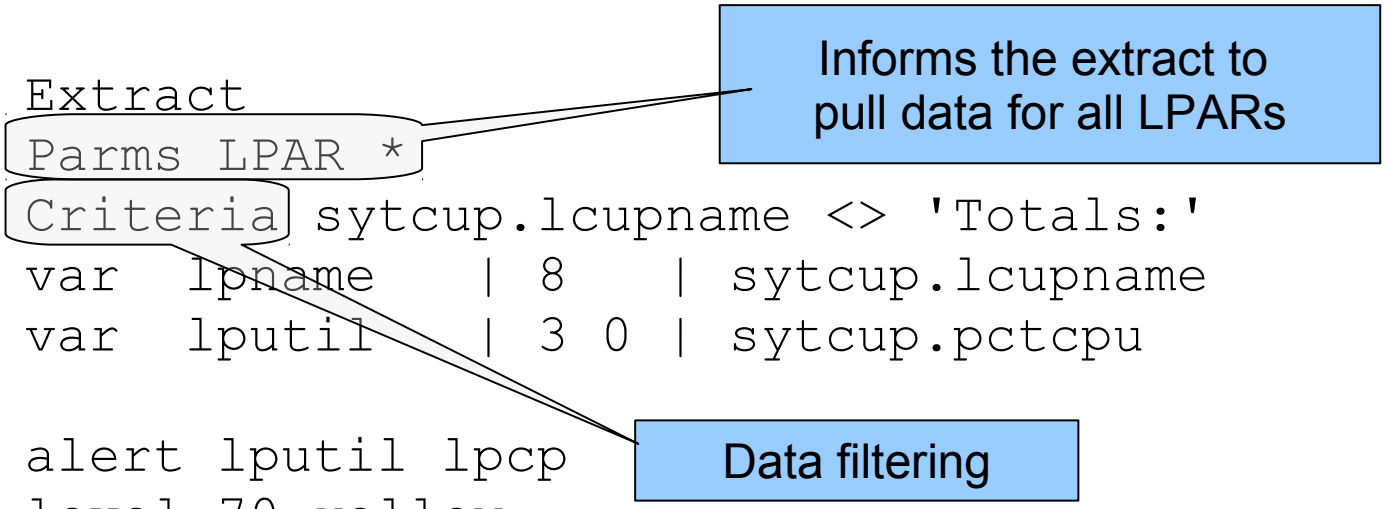
LPAR Utilization

- **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%
```

LPAR Utilization

- **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%
```

- **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)

LPAR Utilization

- **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.vmdttime > 0
var    userid      | 8    | userdata.userid
var    cpuutil     | 3 1  | useact.vmdttime * 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    vmdttime    | 5 2  | useact.vmdttime
```

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---> AddSpc Pages	VDSK Blks	<--pages--> Resi- dent	Lock- ed	Prv or Shr	VIO rate /min	Usr Lks	<AddSpce> Cre- ates	Del- etes	<----------pages/s Sto- len	<--DASD---> 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

Select address spaces beginning with vdisk

Common second virtual 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.qdiiocnt

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
```

Notifications

- **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

Notifications

- **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 &atext
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)
```

Notifications

- **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)
  SPOL Spool utilization is 72% (above 70)
  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)		

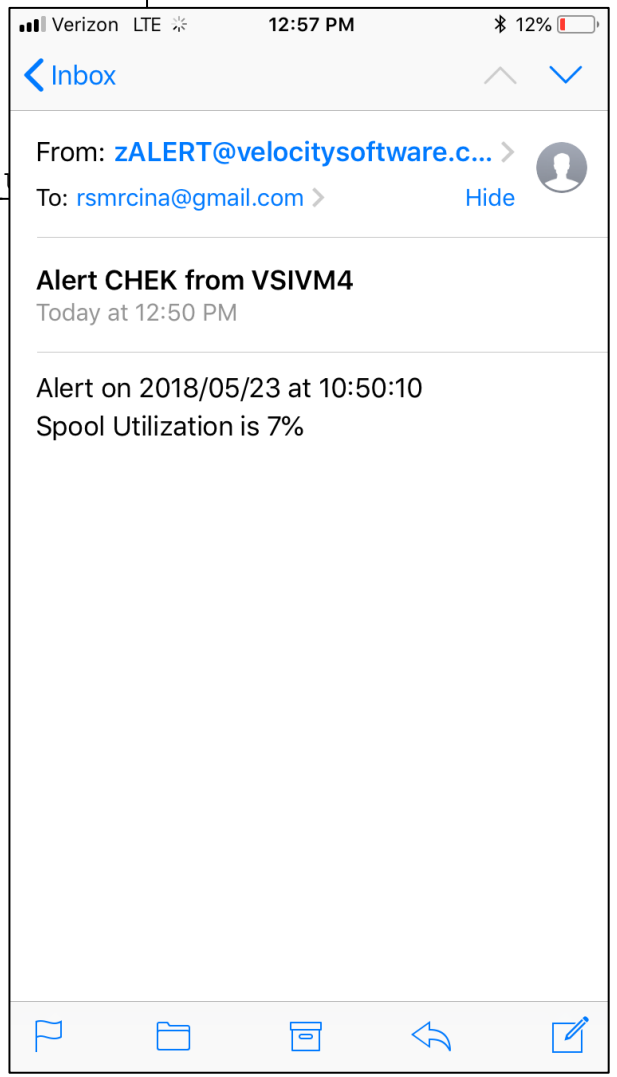
Notifications

- **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
```

```
 VSIVM4   NOTIFY   C1  F 80

      |...+....1....+....2
===== * * * Top of File * * *
===== *
===== * * * End of File * * *
```



VELOCITY SOFTWARE

Alert Options

- **Options add additional function at the alert level**
  - Priority
  - Log
  - Count
  - Separated on alert directive with a vertical bar

```
ALERT DISKPCT LNDX | <options>
```

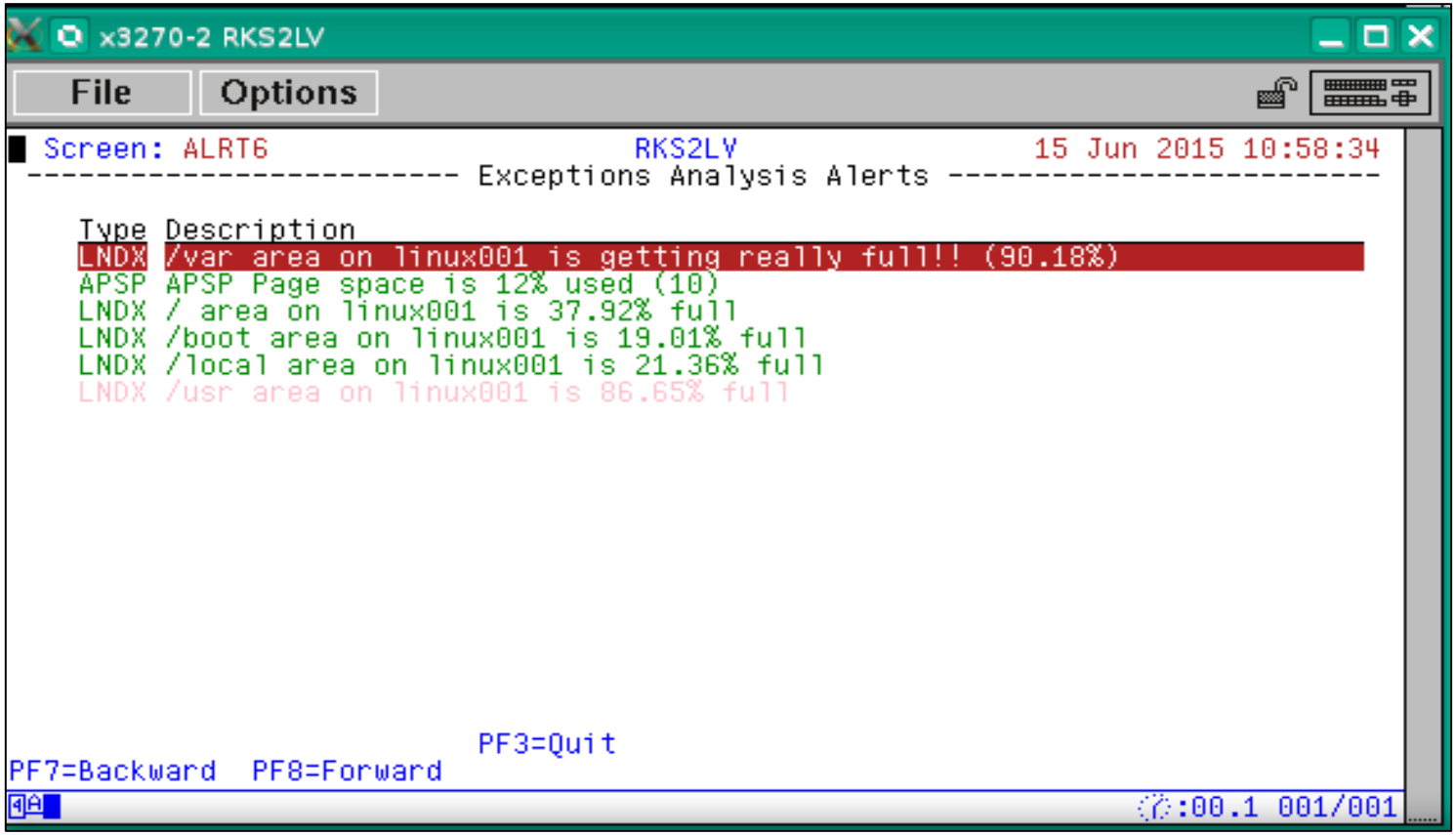
- **Log - Writes alert text displays to a file**
- **Count - Tallies the number of times an alert appears on the display**

Priority

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

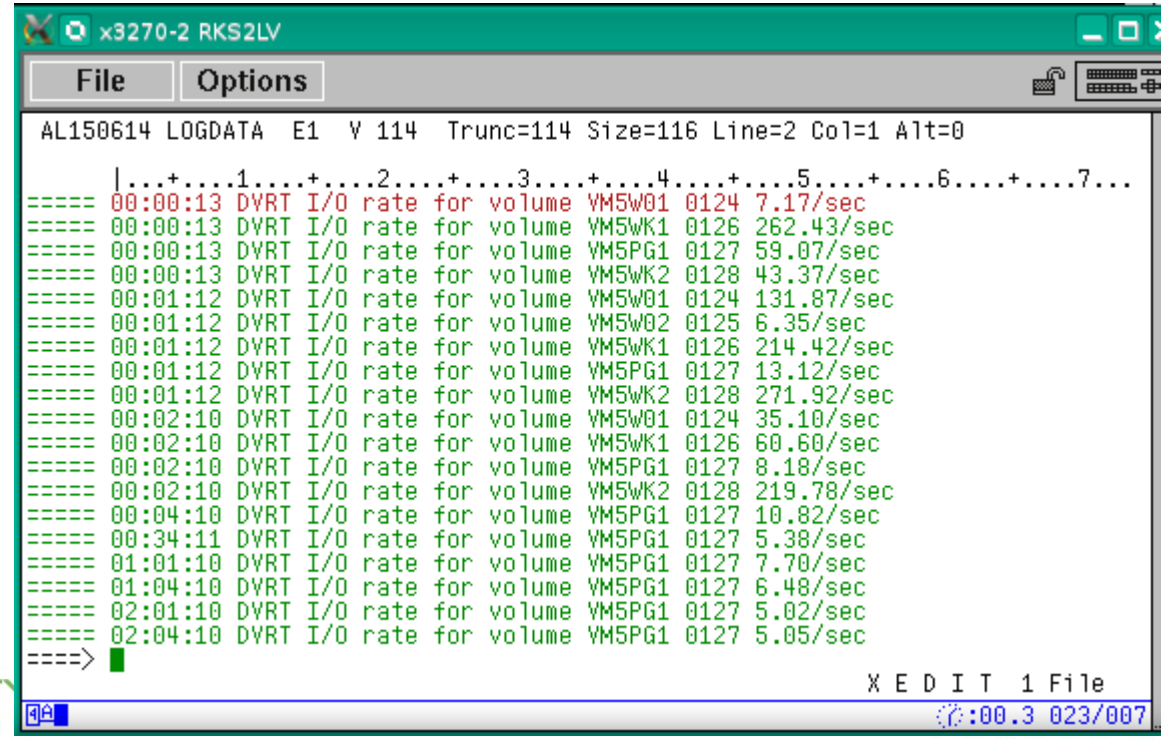
```
extract
parms node *
criteria hstmem.used > 0
var    node     | 8    | tcpip.node
var    memused  | 6 2  | (hstmem.used/hstmem.size)*100
var    desc     | 16   | hstmem.desc
function diskpct  | 6 2 | &node &memused &desc

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

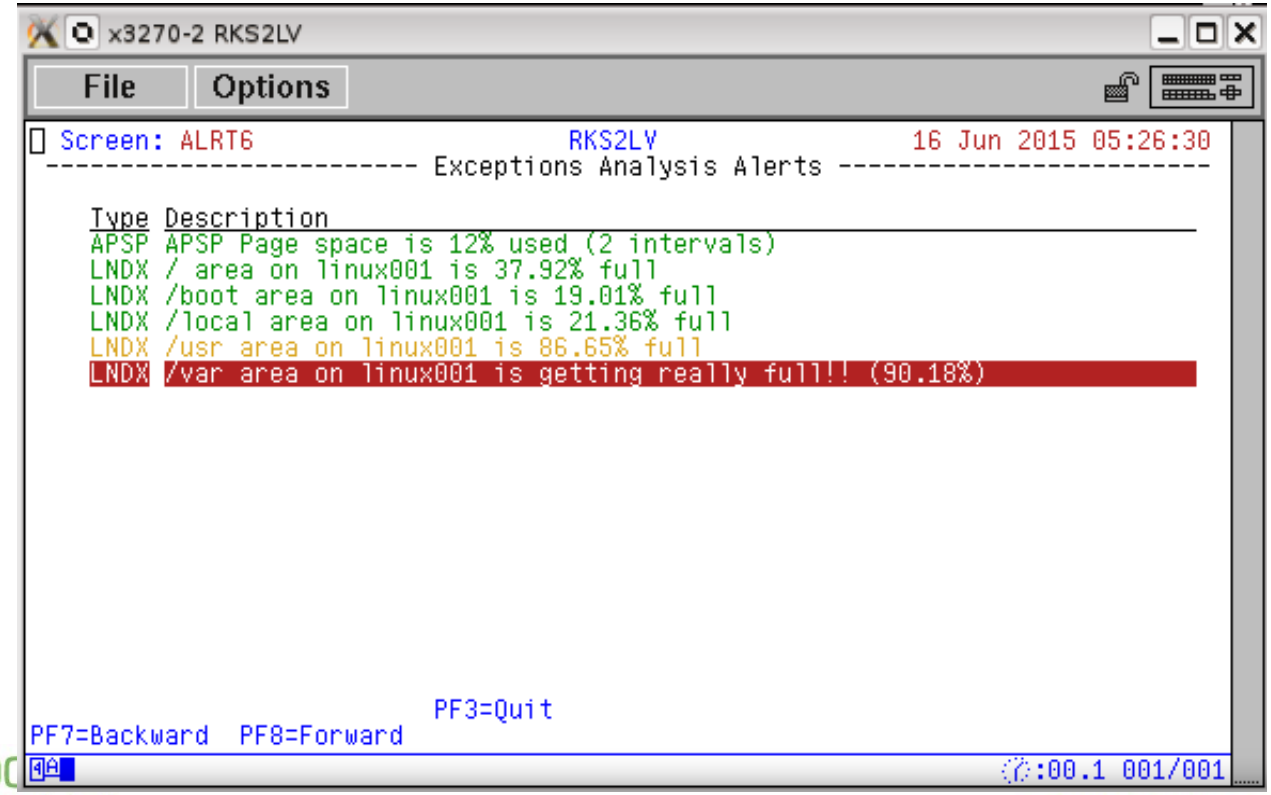
Log

```
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
```



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)
```



Alert Options

- **Log writes out the currently displayable text message**
  - TEXT or LTEXT
- **Count takes an optional key value**
  - Used when an alert can return multiple values
  - Eg: user, node, device
  - Specify variable that contains the key value after COUNT keyword

alert usercpu vmcp | count &userid

- **Multiple options can be specified**

Enable/Disable

- **Disable removes an alert from evaluation (4.2)**
  - 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 apsp
disable
level 10  green
level 30  yellow
level 50  red
text Page space is &page_use% used
```

Limit

- **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%
```

Limit

- **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%
```

Key field

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

Number of intervals to delay executing ACTION

Limit

- **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  | (sytasg.calslti2*100)/sytasg.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

External Processing

- **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

External Processing

- **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 &node is at &diskpct%
level 20  green
level 50  yellow
level 80  pink
level 90  red rev
```

Size of returned value

Parameters passed as exec args

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

- **REXX exec called from the alert**

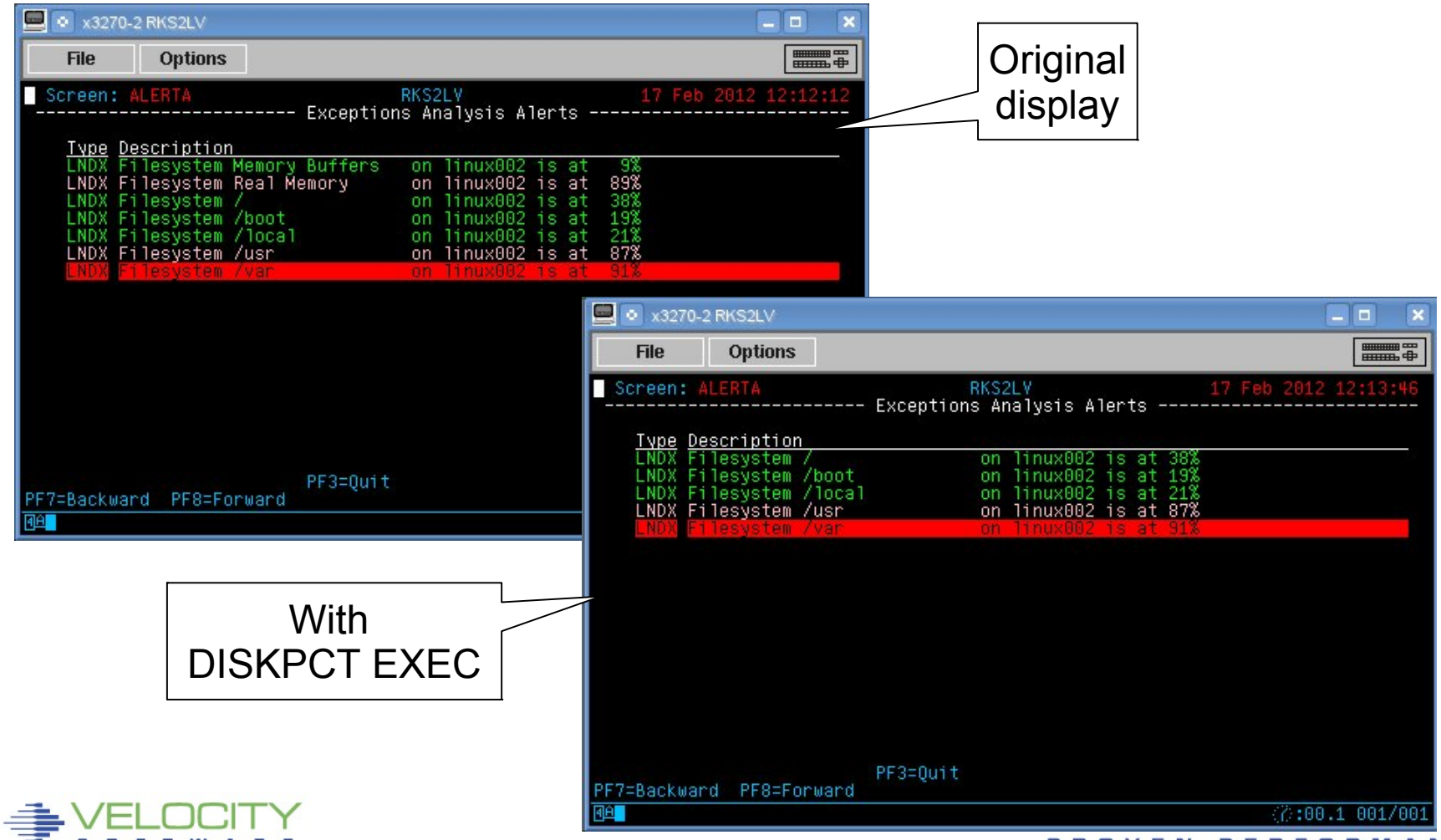
```
/* DISKPCT EXEC: Filter out memory or read-only filesystems */
parse arg node pct descr .

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

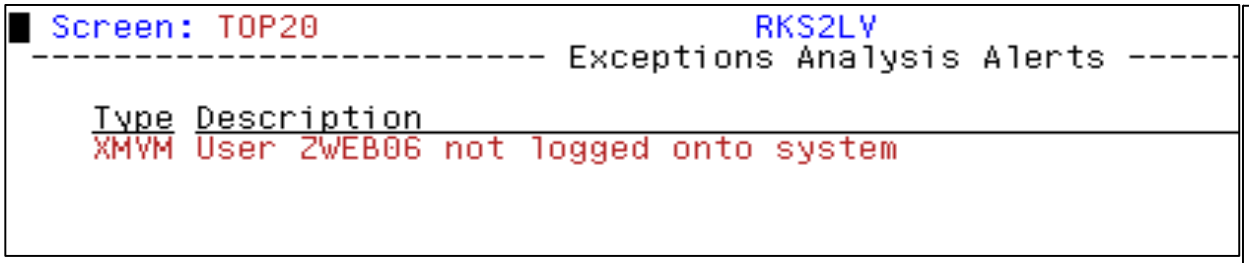
External Processing

- **Results of function call**



- **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 &atext
text User &alrtmusr 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
/*
CRON
```

z/OS alerts

- **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    | zossid.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
```

MongoDB alerts

- **Same for Mongo, Docker and ILMT**

```
extract
parms node *
criteria vsimng.seconds > 0
var userid    | 8    | tcpip.node
var upcount   | 8 0  | VSIMNG.GUPDATE+VSIMNG.GDELETE+VSIMNG.GINSERTS
var qucount   | 8 0  | VSIMNG.GQUERY+VSIMNG.GGETMORE+VSIMNG.GCOMMAND

alert upcount mnou
level 5 yellow
text User &userid Mongo update count &upcount

alert qucount mnqu
level 400 yellow
level 800 red
text User &userid Mongo query count &qucount
```

```
Screen: MONGO          Velocity Software - VSIVM4       17 Jun 2019 09:17:14
------------------------ Exceptions Analysis Alerts ------------------------

  Type Description
  MNOU User mongo01 Mongo update count 12589
  MNQU User mongo01 Mongo query count 49614
```

TCP/IP (customer request)

- **Failed attempts**

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

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 4400

- **4400 very much a 'clean up' release**
  - Details in the README
- **Notification feature accepts group email list**
  - `@filename.filetype` identifies a file that contains a list of email addresses
- **MAILFROM config option**
- **NTFYSUBX notification exit**
  - Sample provided

Updates in 4400

- **EVERY option evaluates an alert at specified intervals**
- **Primary use is for alerts with external routines that don't need to be run each interval**

```
extract
var   poolnm      | 8    |
var   poolgrp     | 3    |
var   poolpct     | 3 0  |
stage sfsmdisk    | 60   | vmsysvps &poolnm &poolgrp &poolpct

alert poolpct sfsm | every 10
level 70 yellow
level 80 red
text Filepool &poolnm Group &poolgrp at &poolpct%
```

Updates in 4400

- **AT option evaluates an alert only specified times**
  - Expressed as minutes past the hour

```
alert sputil spol | at 0:10:15:20:30:40:45:50
level  0  green
level 50  yellow
level 80  red
text Spool utilization is &sputil%
```

Coming up in 5100 (hopefully)

- **Expand wildcard support for include/exclude**
- **Value difference alert**
  - Calculates the difference from current value and previous value
  - Very useful for page/spool utilization
- **RELOAD equivalent to REREAD**
- **Common key definition**
- **Extract name definition**
- **Configurable stale count**

Questions

Rich Smrcina
Velocity Software, Inc
rich@velocitysoftware.com