Going the Last Mile with HP OpenView VantagePoint Operations: Advanced Template Customization

Greg Vaidman
Senior Consultant
Melillo Consulting, Inc.

Why?

- Decrease redundant and extraneous messages
- Make existing messages more detailed and relevant (root cause analysis)

→Make your staff more productive

Agenda

- **New VPO features for advanced customization**
- Write your own monitors, commands and actions
- Customizations to the default templates
- Message correlation
- Other ideas

New Customization Features in ITO 5.x

- Command-line template distribution via *opcragt*
- Command-line tool *opcnode*
- Filtering Internal ITO Messages

New Customization Features in VPO 6.x

- Message Keys/Correlation
- Command-line tool *itotemplate*

Agenda

- New VPO features for advanced customization
- **Write your own monitors, commands and actions**
- Customizations to the default templates
- Message correlation
- Other ideas

Writing Scripts for use with VPO

- “Source” in **`/opt/OV/bin/ov.envvars.sh`**
- Set your PATH explicitly
- Code for multiple architectures/OS’s
- ‘**`exit 0`**’ at the end
- Error checking, error checking, error checking!

- Examples coming up later!

Enabling Scripts for Actions, Commands and Monitors

- Copy the file to:
  `/var/opt/OV/share/databases/OpC/mgd_node/customer/<vendor>/<hardware>/<OS>/`
  - `/actions`
  - `/cmds`
  - `/monitors`

Agenda

- New VPO features for advanced customization
- Write your own monitors, commands and actions
- **Customizations to the default templates**
- Message correlation
- Other ideas

Default UNIX Logfile Templates

	AIX	HP-UX	IRIX	Linux	Solaris
Audit Log	X				
Bad Logs	X	X	X		X
Boot		X			
Cron		X	X	X	X
Kernel Logs	X	X			X
Logins	X	X	X	X	X
Mailqueue		X			
Messages				X	
Su	X	X	X		X
Syslog	X	X	X		X

The Cron Template Problem

- VPO reads logfiles one line at a time

but

- A single job logs multiple entries in the cron log
- When a job fails, the log only tells you the PID
- You have to look up the script/program name

so

- Out of the box: all log entries in the message browser

The Cron Template Solution

- Suppress extraneous messages
  - start of cron job
  - cron job command name
  - successfully completed job
- On failed jobs:
  - Take an automatic action to run a script:
    - Pass the PID as a parameter
    - Find the program name, start and end time and return code
    - Use ‘opcmsg’ to generate a new message with this info
  - Automatically acknowledge the original message

The Cron Template Solution

- Result:
  - One message with only the important info!

The Cron Automatic Action Script: Finding the Job Information (1)

- Use an explicit shell
- Use source code control (RCS, SCCS, etc.)
- Comment liberally

```
#!/usr/bin/ksh
#-----------------------------------------------------------------------
# Author: Greg Vaidman, Melillo Consulting (greg@mjm.com)
# Script: cronfind.sh (determine problem job based on PID from cron log)
# Syntax: cronfind.sh <pid>
# $Header$
#-----------------------------------------------------------------------
```

The Cron Automatic Action Script: Finding the Job Information (2)

- Set the PATH
- Check for the OS: set OS-specific variables

```
. /opt/OV/bin/ov.envvars.sh
export PATH=$OV_BIN/OpC:/usr/bin:/usr/sbin:$PATH

OS=$( /usr/bin/uname -s )
case $OS in
(SunOS)   AWK=/usr/bin/nawk
          LOG=/var/cron/log
          ;;
(HP-UX)   AWK=/usr/bin/awk
          LOG=/var/adm/cron/log
          ;;
esac
```

The Cron Automatic Action Script: Finding the Job Information (3)

- Validate passed parameters

```
# make sure we get a parameter passed
if [[ -z $1 ]]; then
    print "syntax: $( basename $0 ) <PID>"
    exit
else
    PID=$1
fi

# make sure the PID is a number
if print "$PID" | grep -v "^[0-9]*$" > /dev/null
then
    print "invalid PID ($PID)"
    print "syntax: $( basename $0 ) <PID>"
    exit
fi
```

The Cron Automatic Action Script: Finding the Job Information (4)

- Do the work

```
tail -500 $LOG | $AWK -v findpid=$PID '

# find the PID in the log file

. . .

opcmsg sev=warning app=cron obj=job msg_g=Job \
    msg_t="Job [$COMMAND] of user [$USER] $STATUS”

exit 0
```

The Bad Logs/Login/Su Issue

Problem:

- We see everytime someone types a wrong password

Solution:

- Don’t suppress all of these messages!
- Simply set up a duplicate suppression
  - e.g., only report when >=5 instances in 2 minutes

Default UNIX Monitor Templates

	AIX	HP-UX	IRIX	Linux	Solaris
Process Monitors					
Inetd	X	X	X	X	X
Sendmail	X	X	X	X	X
Utilization Monitors					
cpu_util		X	X	X	X
disk_util	X	X	X	X	X
proc_util	X	X	X		X
swap_util	X	X	X	X	X
Other					
MailQueueLength	X	X	X	X	X

Process Monitors

- Make sure that a process is running
  - vp_chk.sh is used by the existing process monitors
  - syntax: **`vp_chk.sh <process_name> <monitor_name>`**
- You can use it too!
- Just copy the “Inetd” monitor & make changes

The Disk Space Monitor Problem

- The default disk space monitor only looks at “/” (root)
- We would need a new monitor for each filesystem!

- Different servers have different filesystems
- We’d need to manually assign each monitor on a per-node basis!

The Disk Space Monitor Solution

- Write a script!
  - Parse the output of ‘df’
  - For each filesystem, run:

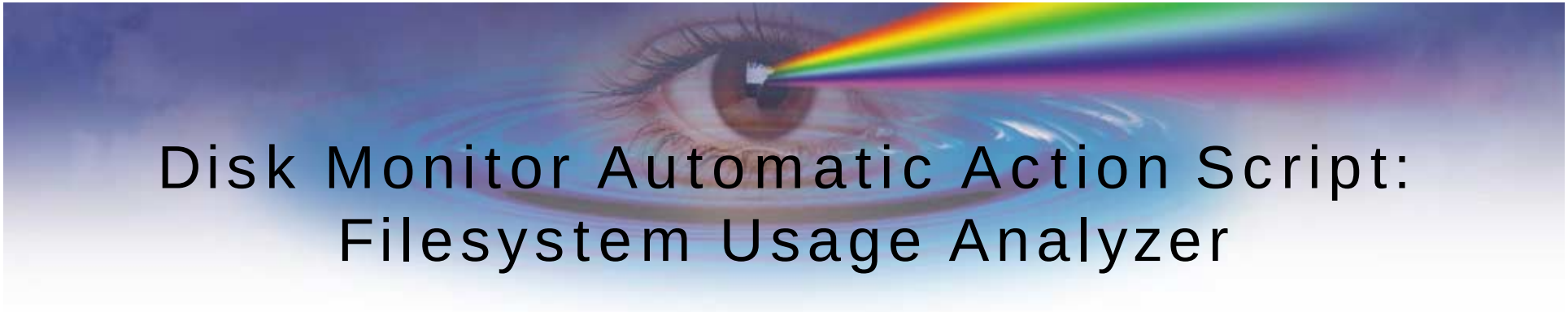
    `opcmon <monitor_name>=<%used> -option=<filesystem>`
- Set up the monitor conditions to use the object field for handling specific filesystems, e.g.,
  - suppress all messages for /cdrom
  - higher thresholds for preallocated database filesystems
  - lower thresholds for more critical filesystems
  - notification/paging only for the most crucial filesystems

Disk Monitor Script: Parsing ‘df’ Output

	Solaris		HP-UX	AIX
	2.5	2.6+		
df options	-lk	-lv	-lp	(none)
df 'filesys' column	6	1	6	6
df '%used' column	5	6	5	5
df header lines	1	1	1	1

Disk Monitor Script: Excerpt

```
/usr/bin/df ${df_opt} | /usr/bin/tail +$(( df_header_lines + 1 )) |
    $AWK -v FCOL=${df_filesys_col} -v PCOL=${df_percent_col} '
          {
                    while (NF<6) {
                              getline nextline
                              $0=$0 " " nextline
                    }
                    filesys=$FCOL
                    percent=$PCOL
                    gsub("%","",percent)
                    print filesys,percent
          }' |
    while read FILESYS PERCENT
    do
          opcmon $MON=$PERCENT -object $FILESYS
    done
```



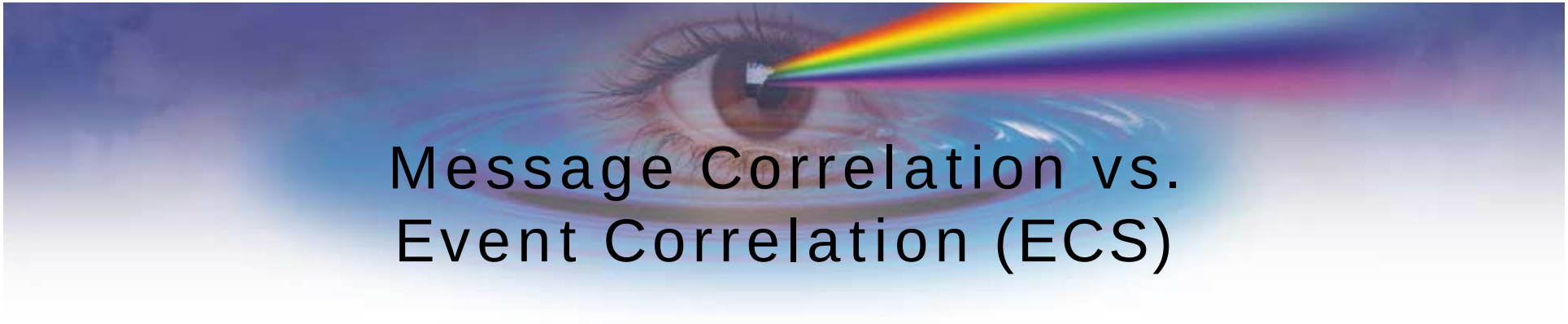
Disk Monitor Automatic Action Script: Filesystem Usage Analyzer

- Create an automatic action for the disk monitor
  - Search the filesystem for large files/directories
    - Warning/Minor: find larger files (>10% of filesystem)
    - Major/Critical: find smaller files (>5% of filesystem)
  - Delete temporary/core files (critical only?)

- Why?
  - The annotation shows what's causing the filesystem to fill up

Agenda

- New VPO features for advanced customization
- Write your own monitors, commands and actions
- Customizations to the default templates
- **Message correlation**
- Other ideas



Message Correlation vs. Event Correlation (ECS)

- Pros
  - Easier to configure
  - Works even after a server restart
  - Built into VPO - no separate product required
- Cons
  - Less flexibility/sophistication
  - No logical relationships (and, or, etc.)
  - only 1-to-many relationships (ECS supports many-to-many)

Message Key

- Gets to the “heart” of the message
  - For the following message text
    - Disk utilization on /usr is 90%
  - The message key definition would be
    - **disk_util:<*$MSG_OBJECT*>:<*$MSG_NODENAME*>:<*$THRESHOLD*>**
  - And this would be the actual message key
    - **disk_util:/usr:www.hp.com:90**

Message Key for Duplicate Suppression

- Previously, duplicate suppression limited to
  - events matching a condition
    - too generic: can't match specific filesystems, processes, etc.
  - identical input events
    - too specific: may contain extraneous info (timestamp, etc.)
- Now:
  - Messages with identical keys are considered duplicates
    - match on just characteristics of the message you want!

Duplicate Suppression

- Example from syslog template:
  - **`Jan 01 13:30:00 ftpd[1234]: User greg: Login incorrect`**
- Use the pattern:
  - **`<@.time> ftpd\\[<#>\\]: User <@.who>: Login incorrect`**
- And a message key of:
  - **`ftp_login_incorrect:<$who>`**

- Suppression is now possible
  - because timestamp is not part of the key
  - only see it if it happens for a particular user >***x*** times in ***y*** minutes

State-Based Browser

- Display only the current info about a resource
  - auto acknowledge old events when new ones appear
  - as fine grained as you want
    - up/down status of a node
    - capacity of a particular filesystem on a particular node
- Cuts down on traffic in the browser!

Message Key Relations for State-Based Browser

- Defines what messages will be replaced
- Defaults can be automatically generated for monitors

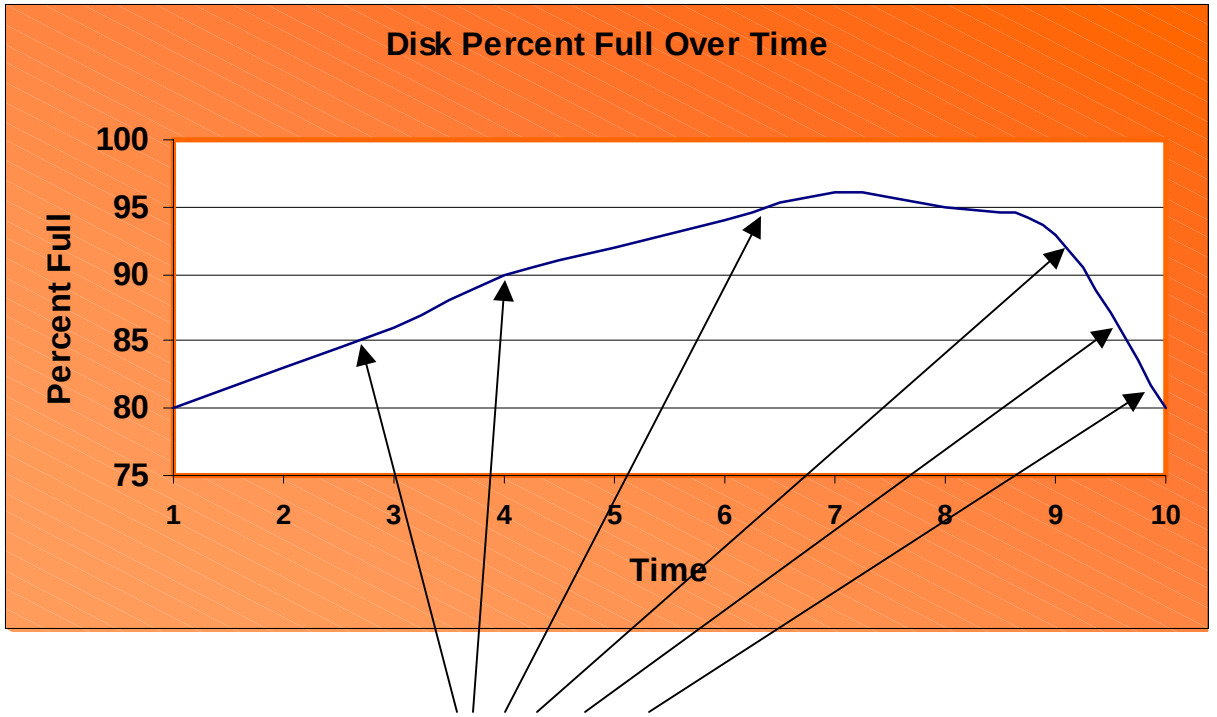
- Message Key for our Disk Monitor
  - **disk_util:<$MSG_OBJECT>:<$MSG_NODENAME>:<$THRESHOLD>**
- Message Key Relationship
  - **disk_util:<$MSG_OBJECT>:<$MSG_NODENAME>:<$*>**

State-Based Browser for Disk Monitor

95% critical (reset @ 92%)

90% major (reset @ 87%)

85% minor (reset @ 82%)



Agenda

- New VPO features for advanced customization
- Write your own monitors, commands and actions
- Customizations to the default templates
- Message correlation
- **Other ideas**

Scripts that Access the ITO Database

- Set Oracle Variables

```
export ORACLE_SID=openview
export ORACLE_BASE=/opt/home/dba/oracle
export ORACLE_HOME=${ORACLE_BASE}/product/8.0.6
export NLS_LANG=american_america.WE8ISO8859P1
export ORA_NLS33=${ORACLE_HOME}/ocommon/nls/admin/data
export SHLIB_PATH=${ORACLE_HOME}/lib
export PATH=${ORACLE_HOME}/bin:$PATH
```

- Use the ‘opcdbpwd’ command

```
$OV_BIN/OpC/opcdbpwd -e sqlplus -s << -EOF-
set echo off;
set pagesize 0;
set linesize 300;
set feedback off;
set heading off;
set trim on;
select count(*) from opc_act_messages;
-EOF-
```

Further Reducing Event Traffic

- Turn on filtering of internal VPO messages
  - put **`OPC_INT_MSG_FLT TRUE`** in opcsvinfo/opcinfo
  - activate duplicate suppression
- Use the duplicate counter in the browser
- Command-line template activation/deactivation
  - e.g., turn off database monitoring during backup
  - avoids “noise”
  - alternative to outage template
  - syntax: **`opctemplate [-e|-d] <`*`template_name`*`>`**

Questions?