IBM Software Group

Automated Performance Management: Strategies, Tips, And Techniques

Ed Woods, IBM Corporation

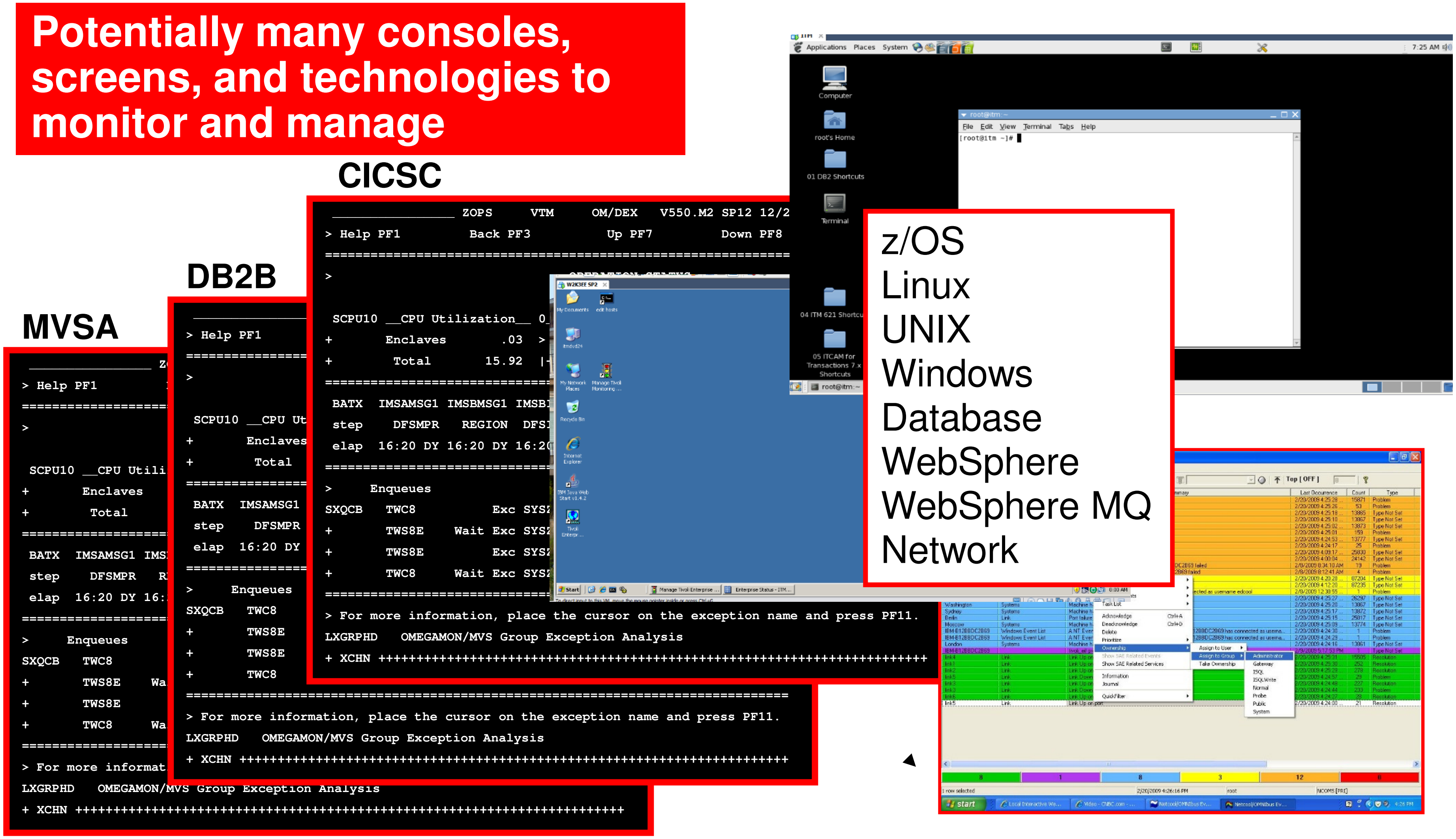
Session #16730
Monday, March 2nd: 4:30 PM - 5:30 PM

Agenda

- Why Integrated Automation?
- Where Automation?
- What is Automated Performance Management?
- Integrated Performance Automation
  - Tools, Tips, Techniques
  - Situations and Policies
- Examples And Best Practices
- Recommendations

Why Integrated Automation?
Islands Of Automation Complicate Management

Potentially many consoles, screens, and technologies to monitor and manage



CICSC

```
_______________ ZOPS      VTM     OM/DEX   V550.M2 SP12 12/2
> Help PF1          Back PF3        Up PF7          Down PF8
=============================================================
>

 SCPU10 __CPU Utilization__ 0
+       Enclaves        .03  >
+        Total        15.92  |
=============================
 BATX  IMSAMSG1 IMSBMSG1 IMSB
 step    DFSMPR   REGION  DFS
 elap  16:20 DY 16:20 DY 16:2
=============================
>      Enqueues
SXQCB    TWC8         Exc SYS
+        TWS8E   Wait Exc SYS
+        TWS8E        Exc SYS
+        TWC8    Wait Exc SYS
=============================
> For more information, place the cursor on the exception name and press PF11.
LXGRPHD    OMEGAMON/MVS Group Exception Analysis
+ XCHN +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

DB2B

```
 ______________
> Help PF1
===============
>

 SCPU10 __CPU Ut
+        Enclaves
+         Total
===============
 BATX  IMSAMSG1
 step    DFSMPR
 elap  16:20 DY
===============
>      Enqueues
SXQCB    TWC8
+        TWS8E
+        TWS8E
+        TWC8
==============================================================================
> For more information, place the cursor on the exception name and press PF11.
LXGRPHD    OMEGAMON/MVS Group Exception Analysis
+ XCHN +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

MVSA

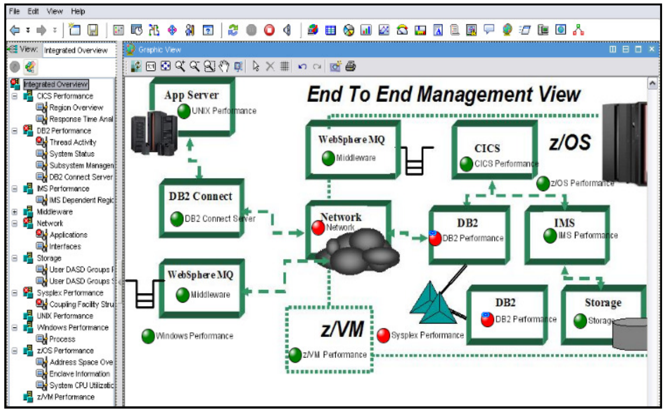
```
 ________________ Z
> Help PF1
===================
>

 SCPU10 __CPU Utili
+       Enclaves
+        Total
===================
 BATX  IMSAMSG1 IMS
 step    DFSMPR   R
 elap  16:20 DY 16:
===================
>     Enqueues
SXQCB    TWC8
+        TWS8E    Wa
+        TWS8E
+        TWC8     Wa
===================
> For more informat
LXGRPHD    OMEGAMON/MVS Group Exception Analysis
+ XCHN +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
[root@itm ~]#
```

z/OS
Linux
UNIX
Windows
Database
WebSphere
WebSphere MQ
Network

Automated Performance Management
Addressing Islands Of Automation

- Many technical platforms, components and core technologies to manage
  - Often each with it’s own group of Subject Matter Experts (SMEs)
  - Potentially with it’s own set of management tools
- The problems
  - Complex SME tools with different User Interfaces
  - SME tools that do not integrate or share information
    - More difficult to navigate
    - More difficult to do problem identification, isolation, and resolution
  - More challenging to automate corrective actions without clearly defined integration



Recommendation – Where feasible pursue a more integrated approach

Where Automation?
Automation Many Occur At Many Levels

- Traditional z/OS console automation
  - Automated resource management
    - System start up and shut down
  - Console message management – message suppression
  - Resource and application management
    - Abend/failure management
    - Subsystem support management
      - WTORs - log management – archive management
- Automation within monitoring and analysis technologies
  - Command and corrective action capabilities within tools
  - Alerts and notifications
- Event/Network management
  - Alerts, notifications and corrective actions managed by the "Manager of Managers" – example Netcool OMNIbus

z/OS console
Address spaces
Messages
Resource status

Monitoring
Resource monitor
Analytics
Real time
History
Alerts – messages

Event Management
Event correlation
Notification
Correction

A Goal For Many Shops
Make Systems Management More ‘Proactive’

- In many shops systems management tends to be done ‘ad hoc’
  - Some alert generation – varies by shop
    - Some shops very alert driven – many are not
  - Often notification consists of ‘call the help desk’
- Many customers want to be more ‘proactive’
  - Definition of proactive may vary
    - Proactive for some installations may mean more rapid alert and notification of technical and/or business application issues
    - Proactive for some installations may mean notification ***prior*** to the problem
      - Alert when utilization indicates a potential issue in the future
      - Alert when I’m within 90% of the wall
    - Proactive may mean an automated workaround or resolution

What Is Automated Performance Management?

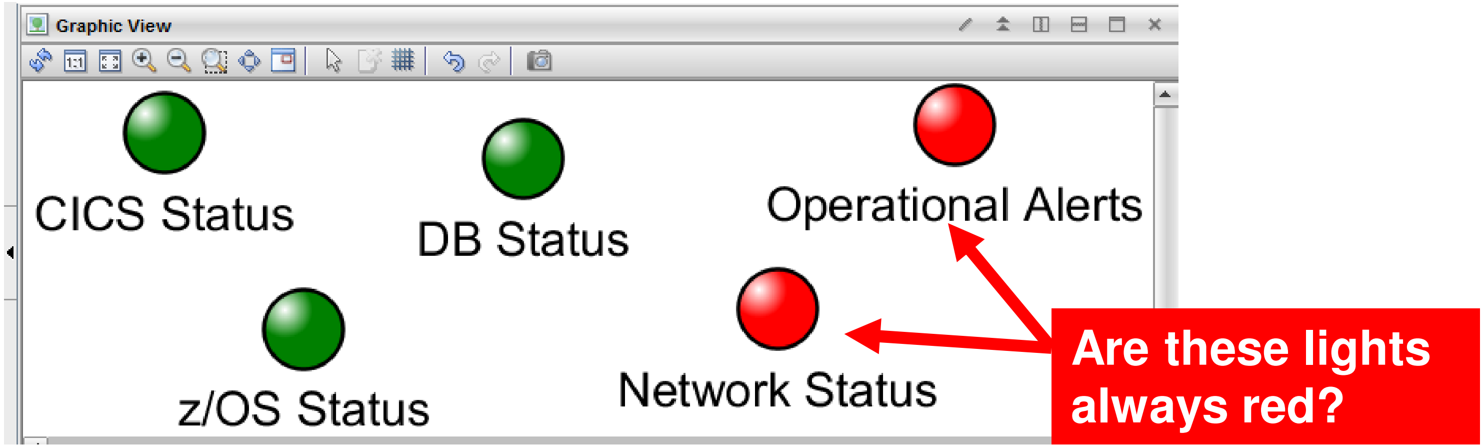
- Exploiting and leveraging the intrinsic monitoring and management capabilities of performance monitoring combined with event management and automation
  - Make automation more powerful and robust
    - Incorporate performance metrics into automation routines
  - Make monitoring more powerful and robust
    - Add message awareness to monitoring tools
  - Incorporate information from the application and/or subsystem performance level
  - Incorporate systems and application knowledge of the staff into automation routines

- The benefits - Become more ‘proactive’
  - Improved and more meaningful/timely alerts and notifications
  - Improved understanding of systems and systems management
  - Reduce the time for problem identification and isolation
  - Improve MTTR (mean time to resolution)
  - Where possible solve problems at machine speed

Benefits Of Automated Performance Management

- Traditional console automation tools focus on console messages and events
- Monitoring information expands the scope of automation
  - Include performance metrics
    - CICS or IMS response time metrics
    - z/OS CPU, paging, and resource utilization metrics
    - Database status and performance metrics
  - Monitoring metrics expand the scope of automation
    - Makes automation more responsive and proactive
      - Avoid application issues and outages
    - Enables more application level automation

Make Alerts More Meaningful
The Challenge Of 'The Always Red Light'



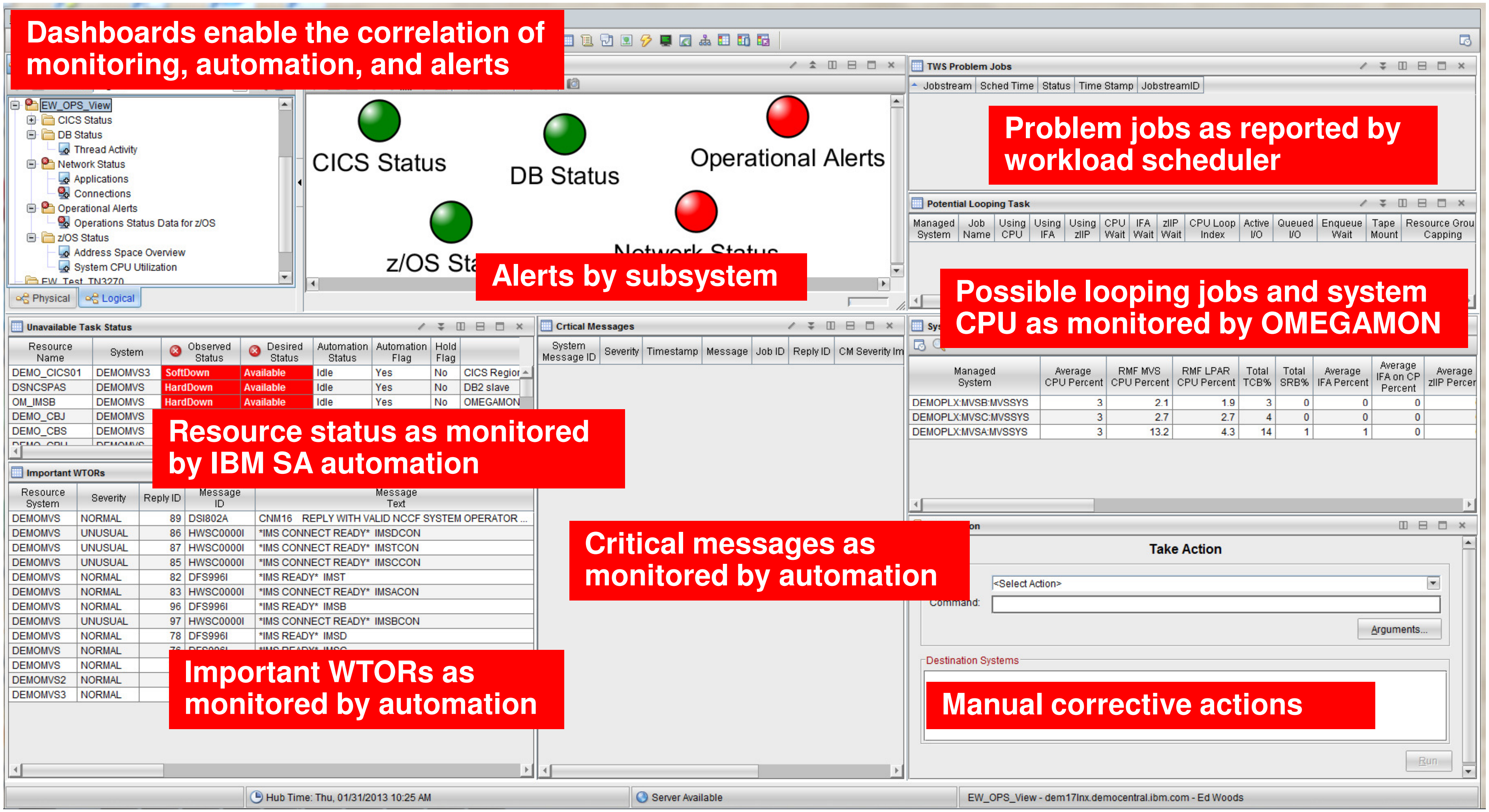
Are these lights always red?

- Alerts should be:
  - Actionable, meaningful, useful
- Integrating automation information with monitoring makes alerts more accurate and meaningful

Alerts
General Recommendations And Rules Of Thumb

- Automation integration helps make alerts Meaningful, Actionable, and Useful
- Meaningful alerts
  - Alert should be flexible – make the names understandable
  - Adopt an alert naming convention
    - Makes it easier to identify customer created versus product provided situations
- Actionable alerts
  - Have appropriate notification
    - A workspace with an alert icon, command/message notification
  - As a standard have expert advice
  - Have pre-defined take actions where appropriate
- Useful alerts
  - Eliminate phony alert indicators – tune out the noise
  - If an alert fires it should indicate an actual issue
    - An alert, an owner, and a consequence

Leverage The Integration Of Automation To Make Alerts And Dashboards More Relevant



Dashboards enable the correlation of monitoring, automation, and alerts

- EW_OPS_View
  - CICS Status
  - DB Status
    - Thread Activity
  - Network Status
    - Applications
    - Connections
  - Operational Alerts
    - Operations Status Data for z/OS
  - z/OS Status
    - Address Space Overview
    - System CPU Utilization
  - EW_Test_TN3270

Alerts by subsystem

TWS Problem Jobs

Jobstream	Sched Time	Status	Time Stamp	JobstreamID

Problem jobs as reported by workload scheduler

Potential Looping Task

Managed System	Job Name	Using CPU	Using IFA	Using zIIP	CPU Wait	IFA Wait	zIIP Wait	CPU Loop Index	Active I/O	Queued I/O	Enqueue Wait	Tape Mount	Resource Grou Capping

Possible looping jobs and system CPU as monitored by OMEGAMON

Unavailable Task Status

Resource Name	System	Observed Status	Desired Status	Automation Status	Automation Flag	Hold Flag	
DEMO_CICS01	DEMOMVS3	SoftDown	Available	Idle	Yes	No	CICS Regior
DSNCSPAS	DEMOMVS	HardDown	Available	Idle	Yes	No	DB2 slave
OM_IMSB	DEMOMVS	HardDown	Available	Idle	Yes	No	OMEGAMON
DEMO_CBJ	DEMOMVS						
DEMO_CBS	DEMOMVS						
DEMO_CPU	DEMOMVS						

Resource status as monitored by IBM SA automation

Crtical Messages

System Message ID	Severity	Timestamp	Message	Job ID	Reply ID	CM Severity Im

Managed System	Average CPU Percent	RMF MVS CPU Percent	RMF LPAR CPU Percent	Total TCB%	Total SRB%	Average IFA Percent	Average IFA on CP Percent	Average zIIP Percer
DEMOPLX:MVSB:MVSSYS	3	2.1	1.9	3	0	0	0	
DEMOPLX:MVSC:MVSSYS	3	2.7	2.7	4	0	0	0	
DEMOPLX:MVSA:MVSSYS	3	13.2	4.3	14	1	1	0	

Important WTORs

Resource System	Severity	Reply ID	Message ID	Message Text
DEMOMVS	NORMAL	89	DSI802A	CNM16 REPLY WITH VALID NCCF SYSTEM OPERATOR ...
DEMOMVS	UNUSUAL	86	HWSC0000I	*IMS CONNECT READY* IMSDCON
DEMOMVS	UNUSUAL	87	HWSC0000I	*IMS CONNECT READY* IMSTCON
DEMOMVS	UNUSUAL	85	HWSC0000I	*IMS CONNECT READY* IMSCCON
DEMOMVS	NORMAL	82	DFS996I	*IMS READY* IMST
DEMOMVS	NORMAL	83	HWSC0000I	*IMS CONNECT READY* IMSACON
DEMOMVS	NORMAL	96	DFS996I	*IMS READY* IMSB
DEMOMVS	UNUSUAL	97	HWSC0000I	*IMS CONNECT READY* IMSBCON
DEMOMVS	NORMAL	78	DFS996I	*IMS READY* IMSD
DEMOMVS	NORMAL	76	DFS996I	*IMS READY* IMSC
DEMOMVS	NORMAL			
DEMOMVS2	NORMAL			
DEMOMVS3	NORMAL			

Important WTORs as monitored by automation

Critical messages as monitored by automation

Take Action

Manual corrective actions

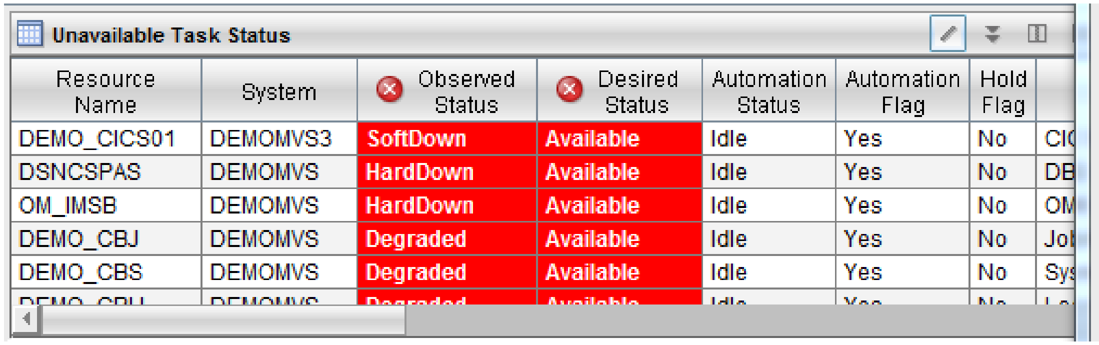
Hub Time: Thu, 01/31/2013 10:25 AM

Server Available

EW_OPS_View - dem17lnx.democentral.ibm.com - Ed Woods

Exploiting The Integration Of Automation And Monitoring

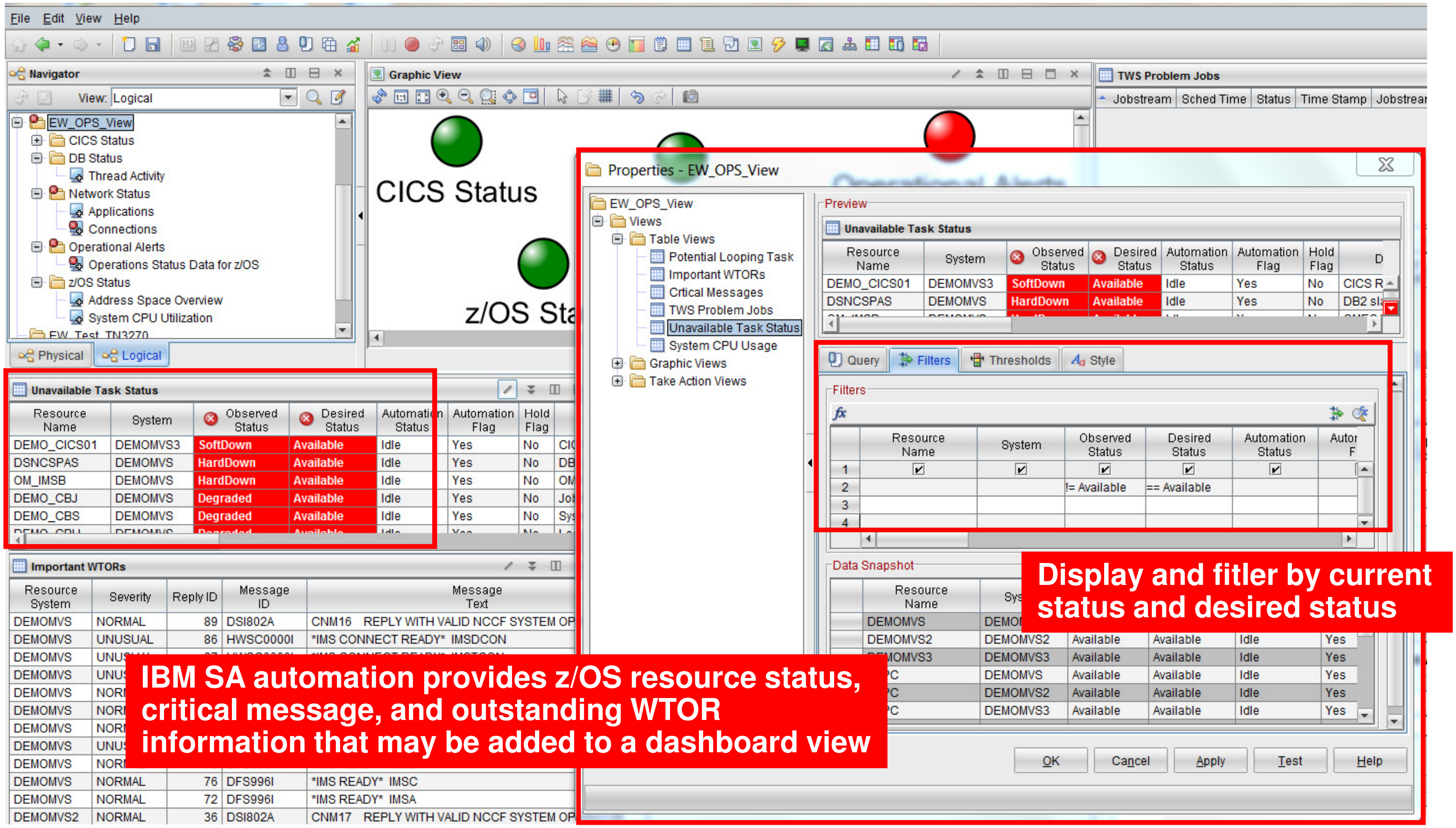
- Leverage automation to make alerts more relevant and useful
  - Automation contains task availability information
    - Current task status, desired status, availability calendar
  - Use to filter out unwanted alerts
    - Example – avoid CICS availability alert during normal outage window
- Leverage automation to expand the scope of alerts
  - Add console message information to alerts
    - Example – Subsystem messages, application error messages, outstanding WTORs



Unavailable Task Status

Resource Name	System	Observed Status	Desired Status	Automation Status	Automation Flag	Hold Flag	
DEMO_CICS01	DEMOMVS3	SoftDown	Available	Idle	Yes	No	CI
DSNCSPAS	DEMOMVS	HardDown	Available	Idle	Yes	No	DB
OM_IMSB	DEMOMVS	HardDown	Available	Idle	Yes	No	OM
DEMO_CBJ	DEMOMVS	Degraded	Available	Idle	Yes	No	Jo
DEMO_CBS	DEMOMVS	Degraded	Available	Idle	Yes	No	Sy
DEMO_CBU	DEMOMVS	Degraded	Available	Idle	Yes	No	L

Consider Resource And Status Information From Automation When Defining A Monitor View



Unavailable Task Status

Resource Name	System	Observed Status	Desired Status	Automation Status	Automation Flag	Hold Flag	
DEMO_CICS01	DEMOMVS3	SoftDown	Available	Idle	Yes	No	CIC
DSNCSPAS	DEMOMVS	HardDown	Available	Idle	Yes	No	DB
OM_IMSB	DEMOMVS	HardDown	Available	Idle	Yes	No	OM
DEMO_CBJ	DEMOMVS	Degraded	Available	Idle	Yes	No	Jol
DEMO_CBS	DEMOMVS	Degraded	Available	Idle	Yes	No	Sys

Important WTORs

Resource System	Severity	Reply ID	Message ID	Message Text
DEMOMVS	NORMAL	89	DSI802A	CNM16 REPLY WITH VALID NCCF SYSTEM OP
DEMOMVS	UNUSUAL	86	HWSC0000I	*IMS CONNECT READY* IMSDCON
DEMOMVS	UNUS			
DEMOMVS	UNUS			
DEMOMVS	NOR			
DEMOMVS	NOR			
DEMOMVS	NOR			
DEMOMVS	UNUS			
DEMOMVS	NOR			
DEMOMVS	NORMAL	76	DFS996I	*IMS READY* IMSC
DEMOMVS	NORMAL	72	DFS996I	*IMS READY* IMSA
DEMOMVS2	NORMAL	36	DSI802A	CNM17 REPLY WITH VALID NCCF SYSTEM OP

Preview – **Unavailable Task Status**

Resource Name	System	Observed Status	Desired Status	Automation Status	Automation Flag	Hold Flag	D
DEMO_CICS01	DEMOMVS3	SoftDown	Available	Idle	Yes	No	CICS R
DSNCSPAS	DEMOMVS	HardDown	Available	Idle	Yes	No	DB2 sl

Filters

	Resource Name	System	Observed Status	Desired Status	Automation Status	Autor F
1	☑	☑	☑	☑	☑	
2			!= Available	== Available		
3						
4						

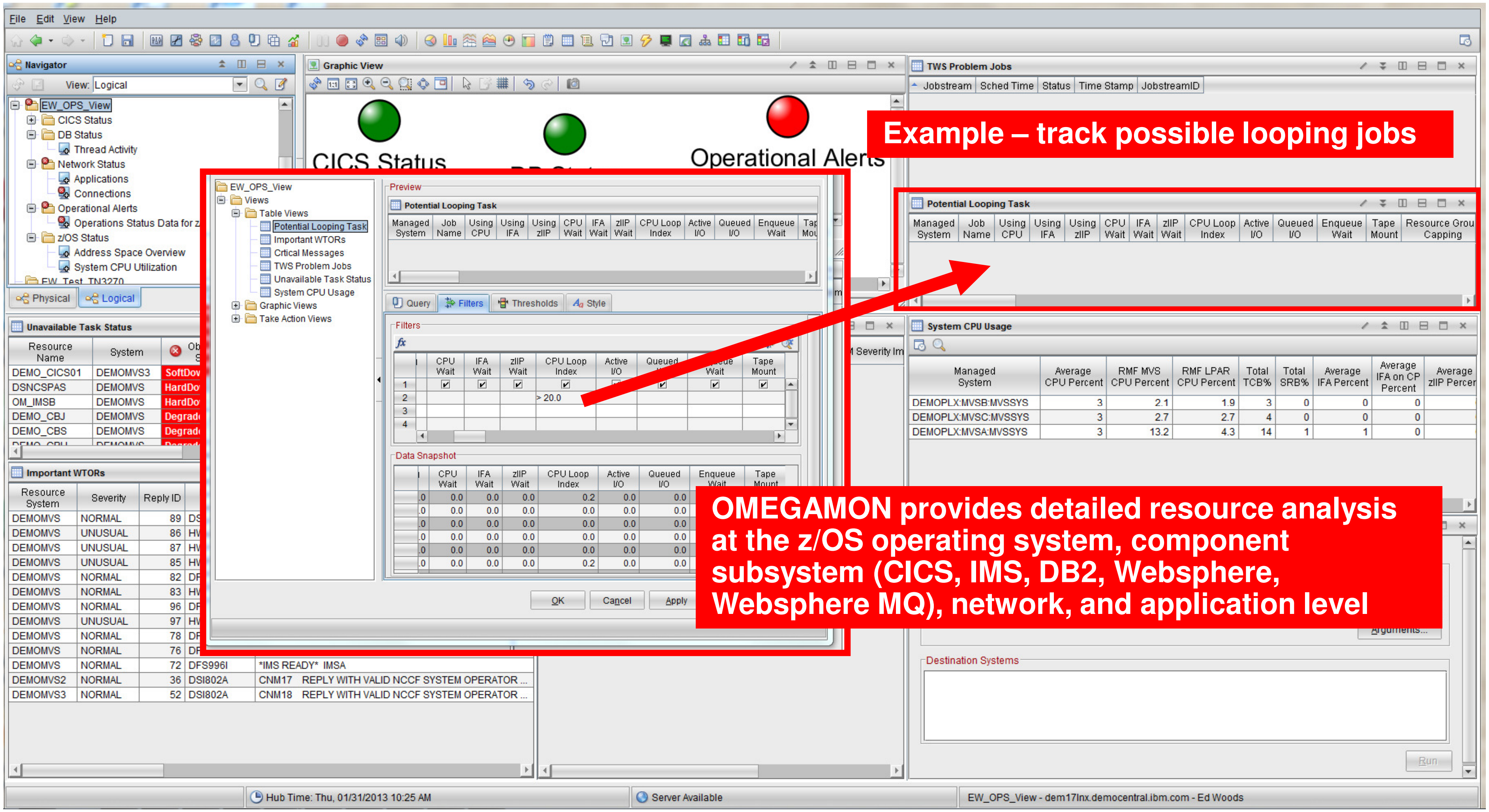
Data Snapshot

Resource Name	Sys				
DEMOMVS	DEMO				
DEMOMVS2	DEMOMVS2	Available	Available	Idle	Yes
MOMVS3	DEMOMVS3	Available	Available	Idle	Yes
C	DEMOMVS	Available	Available	Idle	Yes
C	DEMOMVS2	Available	Available	Idle	Yes
C	DEMOMVS3	Available	Available	Idle	Yes

Display and fitler by current status and desired status

IBM SA automation provides z/OS resource status, critical message, and outstanding WTOR information that may be added to a dashboard view

Integrate With Resource And Analysis Information From z/OS Monitoring (Such As OMEGAMON)



Example – track possible looping jobs

OMEGAMON provides detailed resource analysis at the z/OS operating system, component subsystem (CICS, IMS, DB2, Websphere, Websphere MQ), network, and application level

Potential Looping Task

Managed System	Job Name	Using CPU	Using IFA	Using zIIP	CPU Wait	IFA Wait	zIIP Wait	CPU Loop Index	Active I/O	Queued I/O	Enqueue Wait	Tape Mount	Resource Grou Capping

Unavailable Task Status

Resource Name	System	Ob S
DEMO_CICS01	DEMOMVS3	SoftDov
DSNCSPAS	DEMOMVS	HardDo
OM_IMSB	DEMOMVS	HardDo
DEMO_CBJ	DEMOMVS	Degrad
DEMO_CBS	DEMOMVS	Degrad

Important WTORs

Resource System	Severity	Reply ID		
DEMOMVS	NORMAL	89	DS	
DEMOMVS	UNUSUAL	86	HV	
DEMOMVS	UNUSUAL	87	HV	
DEMOMVS	UNUSUAL	85	HV	
DEMOMVS	NORMAL	82	DF	
DEMOMVS	NORMAL	83	HV	
DEMOMVS	NORMAL	96	DF	
DEMOMVS	UNUSUAL	97	HV	
DEMOMVS	NORMAL	78	DF	
DEMOMVS	NORMAL	76	DF	
DEMOMVS	NORMAL	72	DFS996I	*IMS READY* IMSA
DEMOMVS2	NORMAL	36	DSI802A	CNM17 REPLY WITH VALID NCCF SYSTEM OPERATOR ...
DEMOMVS3	NORMAL	52	DSI802A	CNM18 REPLY WITH VALID NCCF SYSTEM OPERATOR ...

Filters

	CPU Wait	IFA Wait	zIIP Wait	CPU Loop Index	Active I/O	Queued	Wait	Tape Mount
1	☑	☑	☑	☑		☑	☑	☑
2				> 20.0				
3								
4								

Data Snapshot

	CPU Wait	IFA Wait	zIIP Wait	CPU Loop Index	Active I/O	Queued I/O	Enqueue Wait	Tape Mount
.0	0.0	0.0	0.0	0.2	0.0	0.0		
.0	0.0	0.0	0.0	0.0	0.0	0.0		
.0	0.0	0.0	0.0	0.0	0.0	0.0		
.0	0.0	0.0	0.0	0.0	0.0	0.0		
.0	0.0	0.0	0.0	0.0	0.0	0.0		
.0	0.0	0.0	0.0	0.2	0.0	0.0		

System CPU Usage

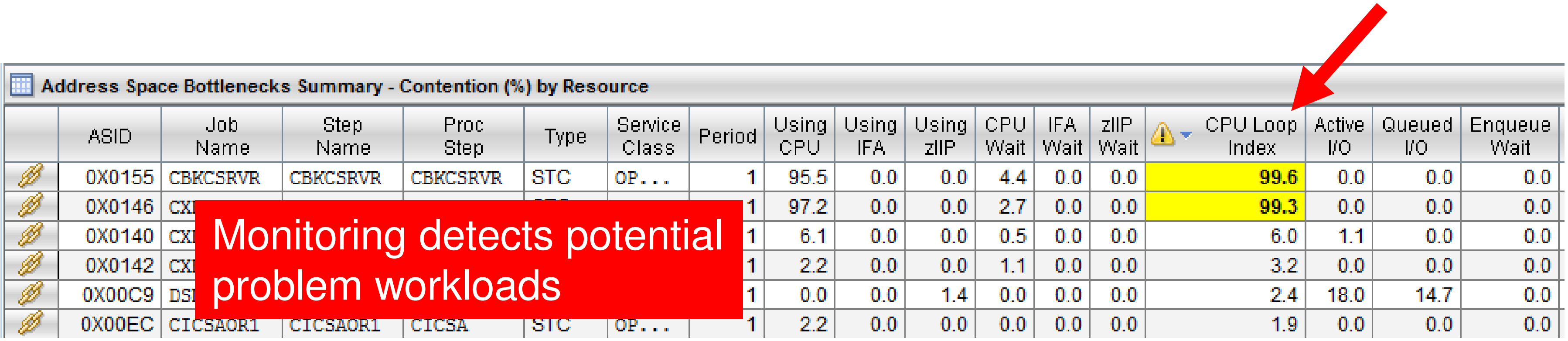
Managed System	Average CPU Percent	RMF MVS CPU Percent	RMF LPAR CPU Percent	Total TCB%	Total SRB%	Average IFA Percent	Average IFA on CP Percent	Average zIIP Percer
DEMOPLX:MVSB:MVSSYS	3	2.1	1.9	3	0	0	0	
DEMOPLX:MVSC:MVSSYS	3	2.7	2.7	4	0	0	0	
DEMOPLX:MVSA:MVSSYS	3	13.2	4.3	14	1	1	0	

Hub Time: Thu, 01/31/2013 10:25 AM

Server Available

EW_OPS_View - dem17lnx.democentral.ibm.com - Ed Woods

Exploit Monitoring To Detect Problem Workload



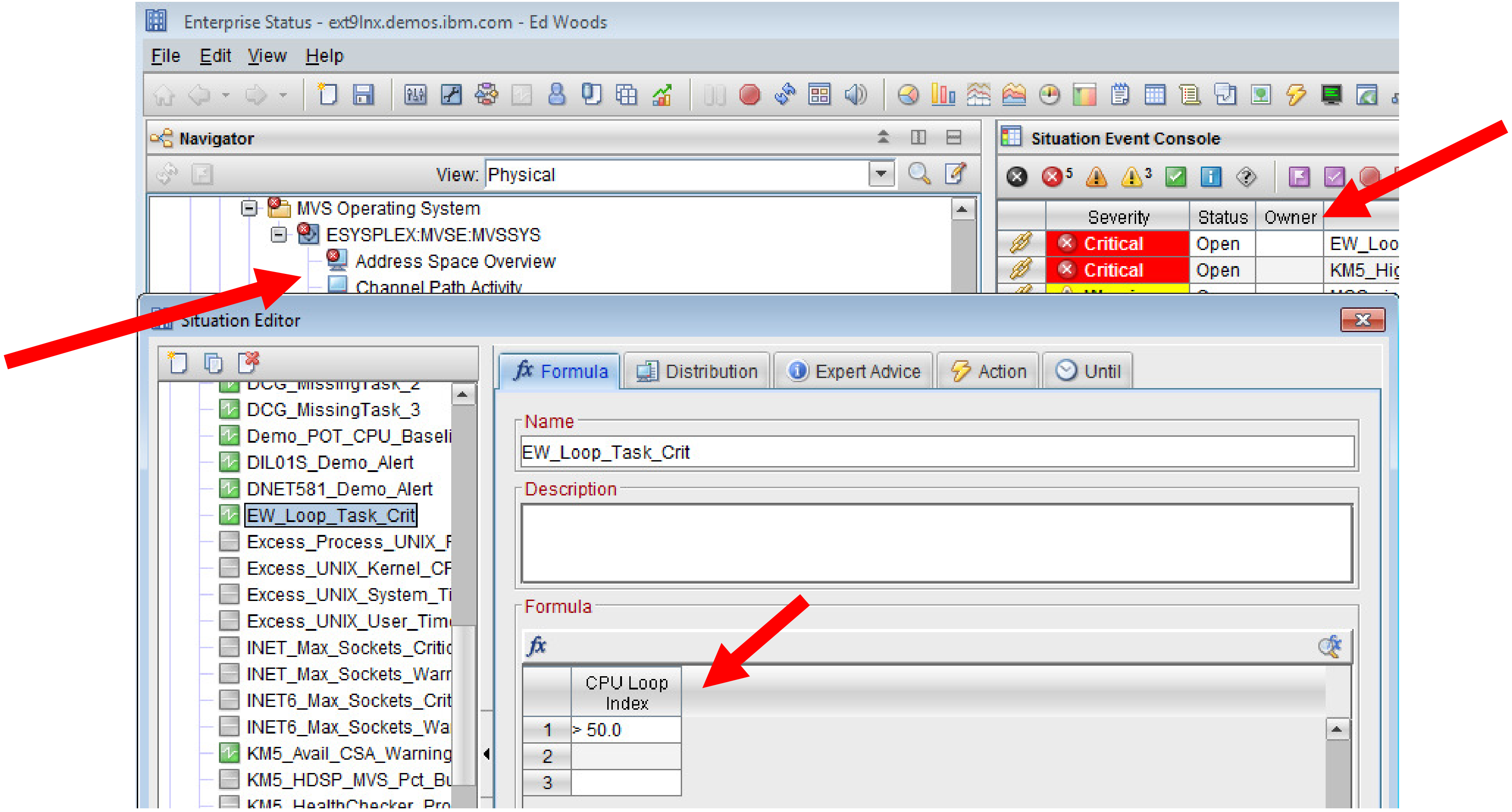
Address Space Bottlenecks Summary - Contention (%) by Resource

	ASID	Job Name	Step Name	Proc Step	Type	Service Class	Period	Using CPU	Using IFA	Using zIIP	CPU Wait	IFA Wait	zIIP Wait	⚠ CPU Loop Index	Active I/O	Queued I/O	Enqueue Wait
	0X0155	CBKCSRVR	CBKCSRVR	CBKCSRVR	STC	OP...	1	95.5	0.0	0.0	4.4	0.0	0.0	99.6	0.0	0.0	0.0
	0X0146	CXI					1	97.2	0.0	0.0	2.7	0.0	0.0	99.3	0.0	0.0	0.0
	0X0140	CXI					1	6.1	0.0	0.0	0.5	0.0	0.0	6.0	1.1	0.0	0.0
	0X0142	CXI					1	2.2	0.0	0.0	1.1	0.0	0.0	3.2	0.0	0.0	0.0
	0X00C9	DSI					1	0.0	0.0	1.4	0.0	0.0	0.0	2.4	18.0	14.7	0.0
	0X00EC	CICSAOR1	CICSAOR1	CICSA	STC	OP...	1	2.2	0.0	0.0	0.0	0.0	0.0	1.9	0.0	0.0	0.0

Monitoring detects potential problem workloads

- Example - OMEGAMON bottle neck analysis may be used to determine potential problem workloads
  - CPU Loop Index detects potential looping tasks
  - CPU Loop Index may be viewed as an ‘analytic’ to score the likelihood of the loop
- Use automation to notify and address looping task issue

Monitor Critical Metrics To Create Useful Alerts



Enterprise Status - ext9lnx.demos.ibm.com - Ed Woods

View: Physical

- MVS Operating System
  - ESYSPLEX:MVSE:MVSSYS
    - Address Space Overview
    - Channel Path Activity

Severity	Status	Owner	
Critical	Open		EW_Loo
Critical	Open		KM5_Hig

- DCG_MissingTask_2
- DCG_MissingTask_3
- Demo_POT_CPU_Baseli
- DIL01S_Demo_Alert
- DNET581_Demo_Alert
- EW_Loop_Task_Crit
- Excess_Process_UNIX_F
- Excess_UNIX_Kernel_CF
- Excess_UNIX_System_Ti
- Excess_UNIX_User_Tim
- INET_Max_Sockets_Critic
- INET_Max_Sockets_Warr
- INET6_Max_Sockets_Crit
- INET6_Max_Sockets_Wa
- KM5_Avail_CSA_Warning
- KM5_HDSP_MVS_Pct_Bu
- KM5_HealthChecker_Pro

Name: EW_Loop_Task_Crit

Description:

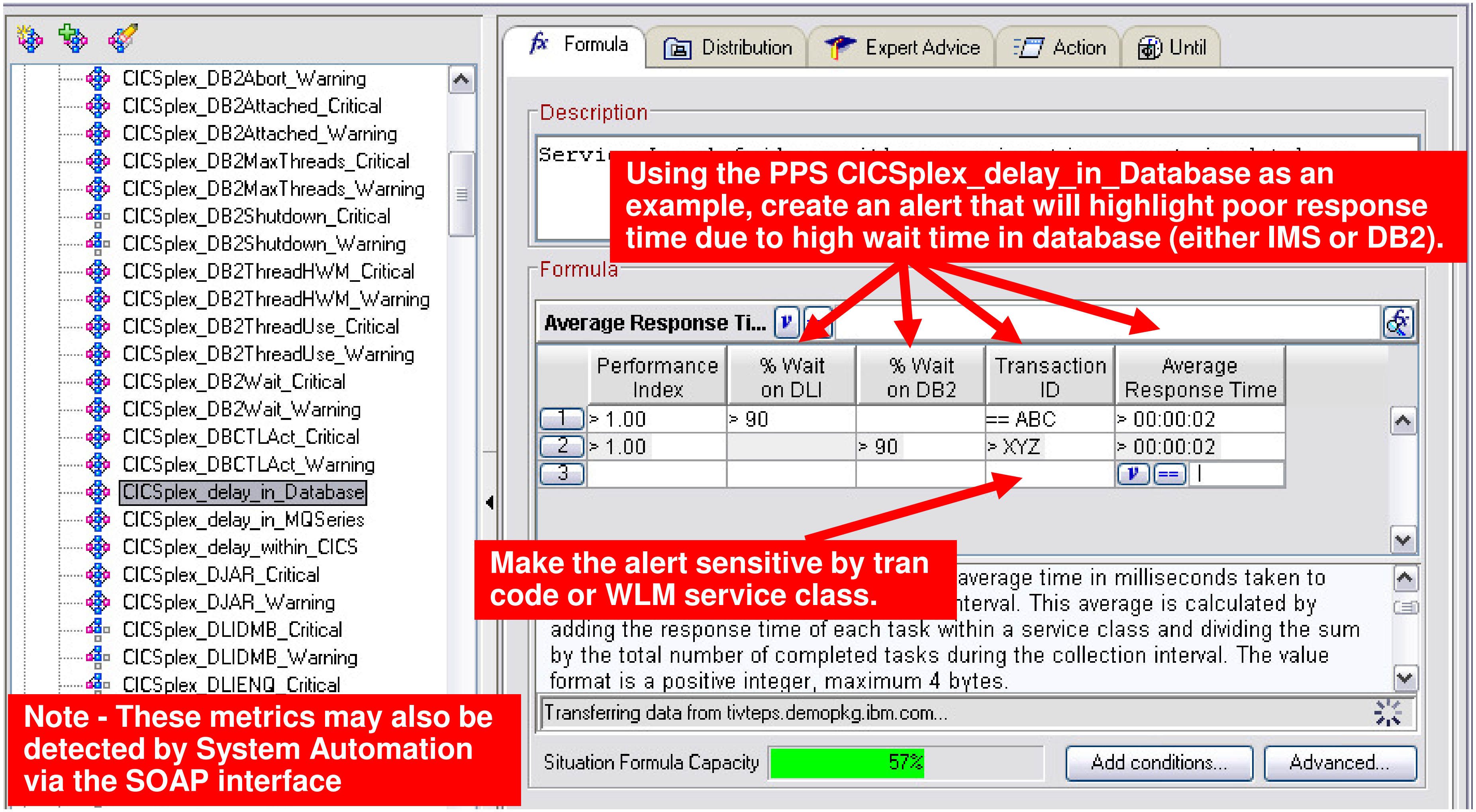
Formula

	CPU Loop Index
1	> 50.0
2	
3	

- Alerts (aka 'Situations') are building blocks to drive further analysis and automation possibilities

CICS Performance Example

Exploit Boolean Logic To Make Alerts More Meaningful



- CICSplex_DB2Abort_Warning
- CICSplex_DB2Attached_Critical
- CICSplex_DB2Attached_Warning
- CICSplex_DB2MaxThreads_Critical
- CICSplex_DB2MaxThreads_Warning
- CICSplex_DB2Shutdown_Critical
- CICSplex_DB2Shutdown_Warning
- CICSplex_DB2ThreadHWM_Critical
- CICSplex_DB2ThreadHWM_Warning
- CICSplex_DB2ThreadUse_Critical
- CICSplex_DB2ThreadUse_Warning
- CICSplex_DB2Wait_Critical
- CICSplex_DB2Wait_Warning
- CICSplex_DBCTLAct_Critical
- CICSplex_DBCTLAct_Warning
- CICSplex_delay_in_Database
- CICSplex_delay_in_MQSeries
- CICSplex_delay_within_CICS
- CICSplex_DJAR_Critical
- CICSplex_DJAR_Warning
- CICSplex_DLIDMB_Critical
- CICSplex_DLIDMB_Warning
- CICSplex_DLIENQ_Critical

Servi

	Performance Index	% Wait on DLI	% Wait on DB2	Transaction ID	Average Response Time
1	> 1.00	> 90		== ABC	> 00:00:02
2	> 1.00		> 90	> XYZ	> 00:00:02
3					

average time in milliseconds taken to nterval. This average is calculated by adding the response time of each task within a service class and dividing the sum by the total number of completed tasks during the collection interval. The value format is a positive integer, maximum 4 bytes.

Transferring data from tivteps.demopkg.ibm.com...

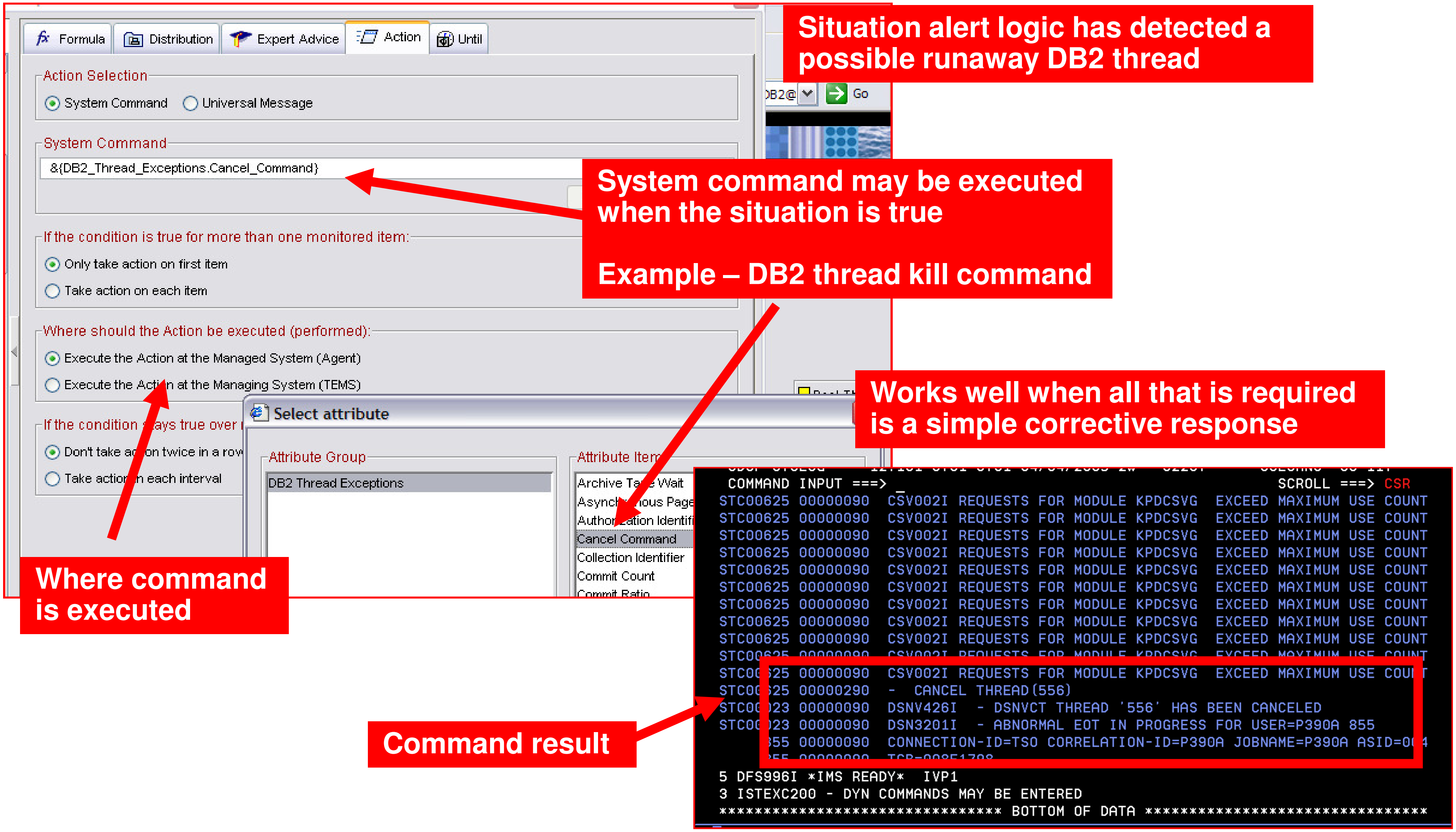
Situation Formula Capacity 57%

Using the PPS CICSplex_delay_in_Database as an example, create an alert that will highlight poor response time due to high wait time in database (either IMS or DB2).

Make the alert sensitive by tran code or WLM service class.

Note - These metrics may also be detected by System Automation via the SOAP interface

Corrective Actions May Be Attached Directly To Alerts
Example - Addressing A “Runaway” DB2 Thread



Action Selection

● System Command ○ Universal Message

System Command

&{DB2_Thread_Exceptions.Cancel_Command}

If the condition is true for more than one monitored item:

● Only take action on first item

○ Take action on each item

Where should the Action be executed (performed):

● Execute the Action at the Managed System (Agent)

○ Execute the Acti n at the Managing System (TEMS)

If the condition ays true over i

● Don't take ac on twice in a row

○ Take action n each interval

Attribute Group

DB2 Thread Exceptions

Attribute Item

Archive Ta e Wait
Asynch ous Page
Author ation Identifi
Cancel Command
Collection Identifier
Commit Count
Commit Ratio

```
 COMMAND INPUT ===> _                                              SCROLL ===> CSR
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000290  -  CANCEL THREAD(556)
STC00 23 00000090  DSNV426I  - DSNVCT THREAD '556' HAS BEEN CANCELED
STC00 23 00000090  DSN3201I  - ABNORMAL EOT IN PROGRESS FOR USER=P390A 855
      355 00000090  CONNECTION-ID=TSO CORRELATION-ID=P390A JOBNAME=P390A ASID=00 4

5 DFS996I *IMS READY*  IVP1
3 ISTEXC200 - DYN COMMANDS MAY BE ENTERED
******************************** BOTTOM OF DATA ********************************
```

Situation alert logic has detected a possible runaway DB2 thread

System command may be executed when the situation is true

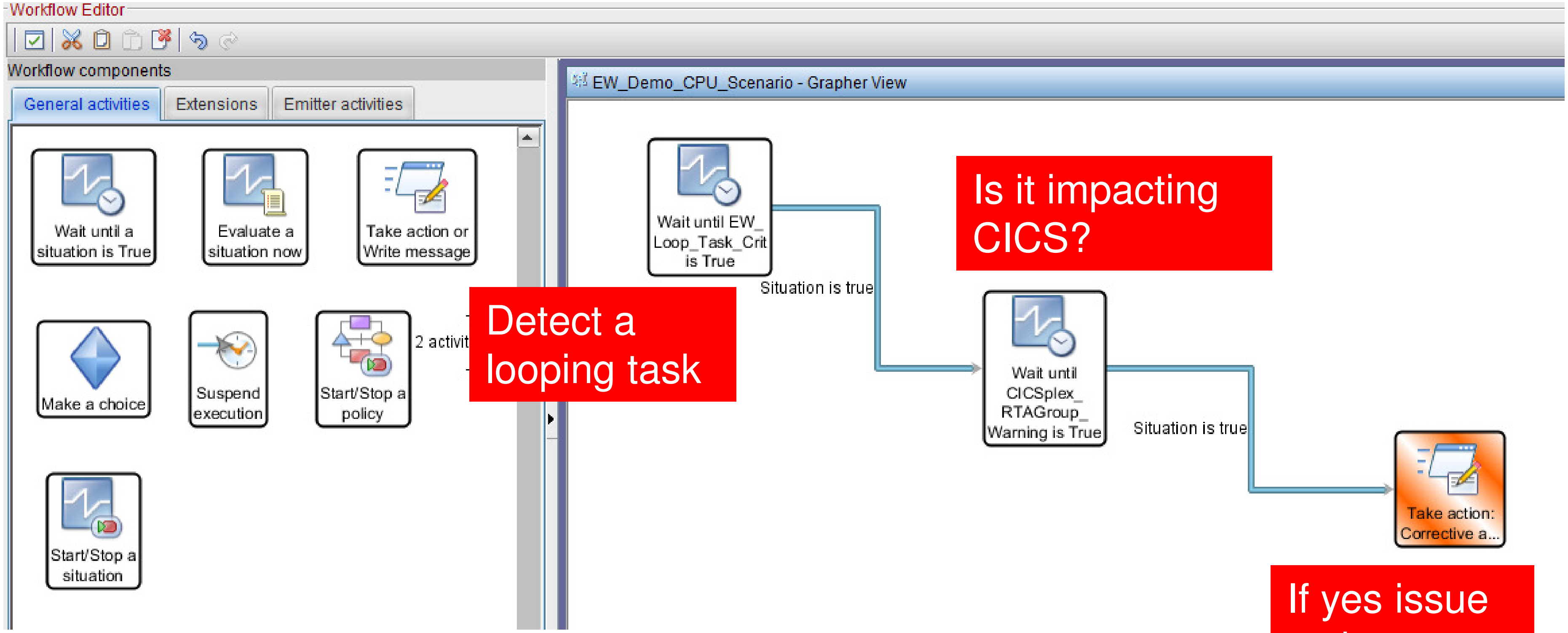
Example – DB2 thread kill command

Works well when all that is required is a simple corrective response

Where command is executed

Command result

More Sophisticated Automation Scenarios May Require A More Detailed Approach



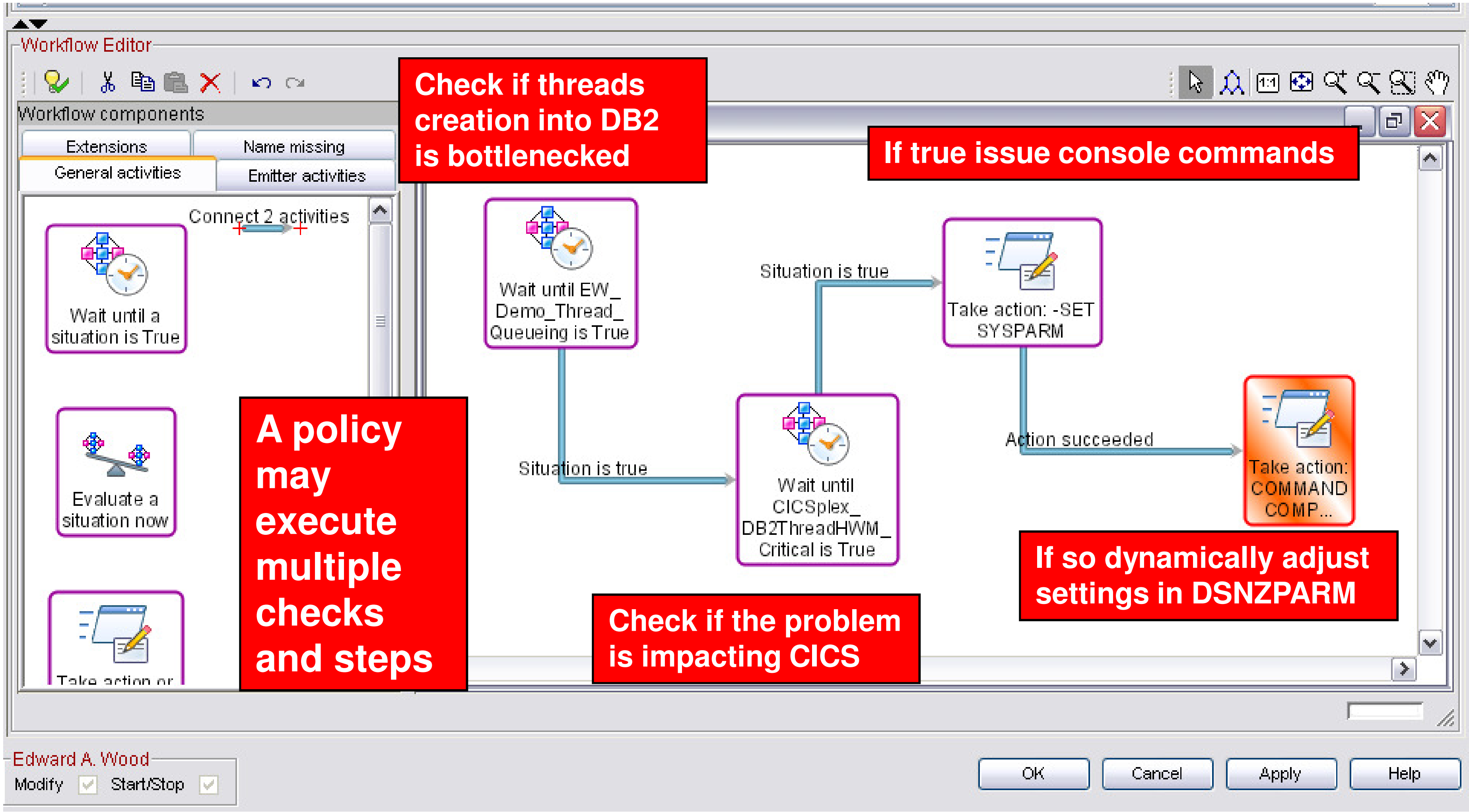
Detect a looping task

Is it impacting CICS?

If yes issue action

- Policy automation enables the easy integration of monitoring and automation
- If there is a problem workload
  - Determine the impact of the issue before executing corrective action

Alert Correlation Via A Policy Mechanism Expands The Concept Of Automated Performance Management



Check if threads creation into DB2 is bottlenecked

If true issue console commands

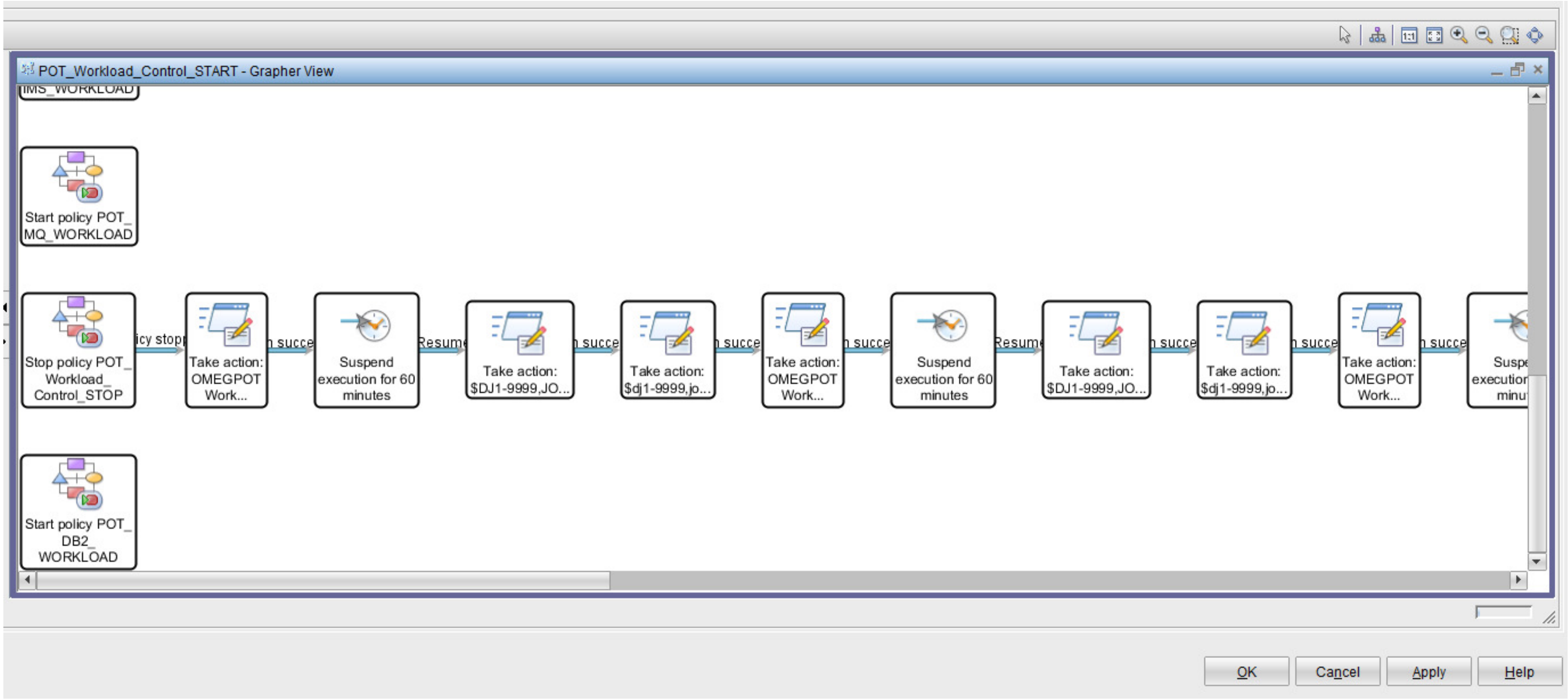
A policy may execute multiple checks and steps

Check if the problem is impacting CICS

If so dynamically adjust settings in DSNZPARM

Using Policies To Automate Multiple Components

Example – Using Policy Automation To Manage PoT Workloads


POT_Workload_Control_START - Grapher View
IMS_WORKLOAD
Start policy POT_ MQ_WORKLOAD
Stop policy POT_ Workload_ Control_STOP
icy stop
Take action: OMEGPOT Work...
succe
Suspend execution for 60 minutes
Resume
Take action: $DJ1-9999,JO...
succe
Take action: $dj1-9999,jo...
succe
Take action: OMEGPOT Work...
succe
Suspend execution for 60 minutes
Resume
Take action: $DJ1-9999,JO...
succe
Take action: $dj1-9999,jo...
succe
Take action: OMEGPOT Work...
succe
Suspe execution minu
Start policy POT_ DB2_ WORKLOAD
OK
Cancel
Apply
Help

- Policy automation may be applied to various components and platforms
- Monitor, manage and maintain complex workloads

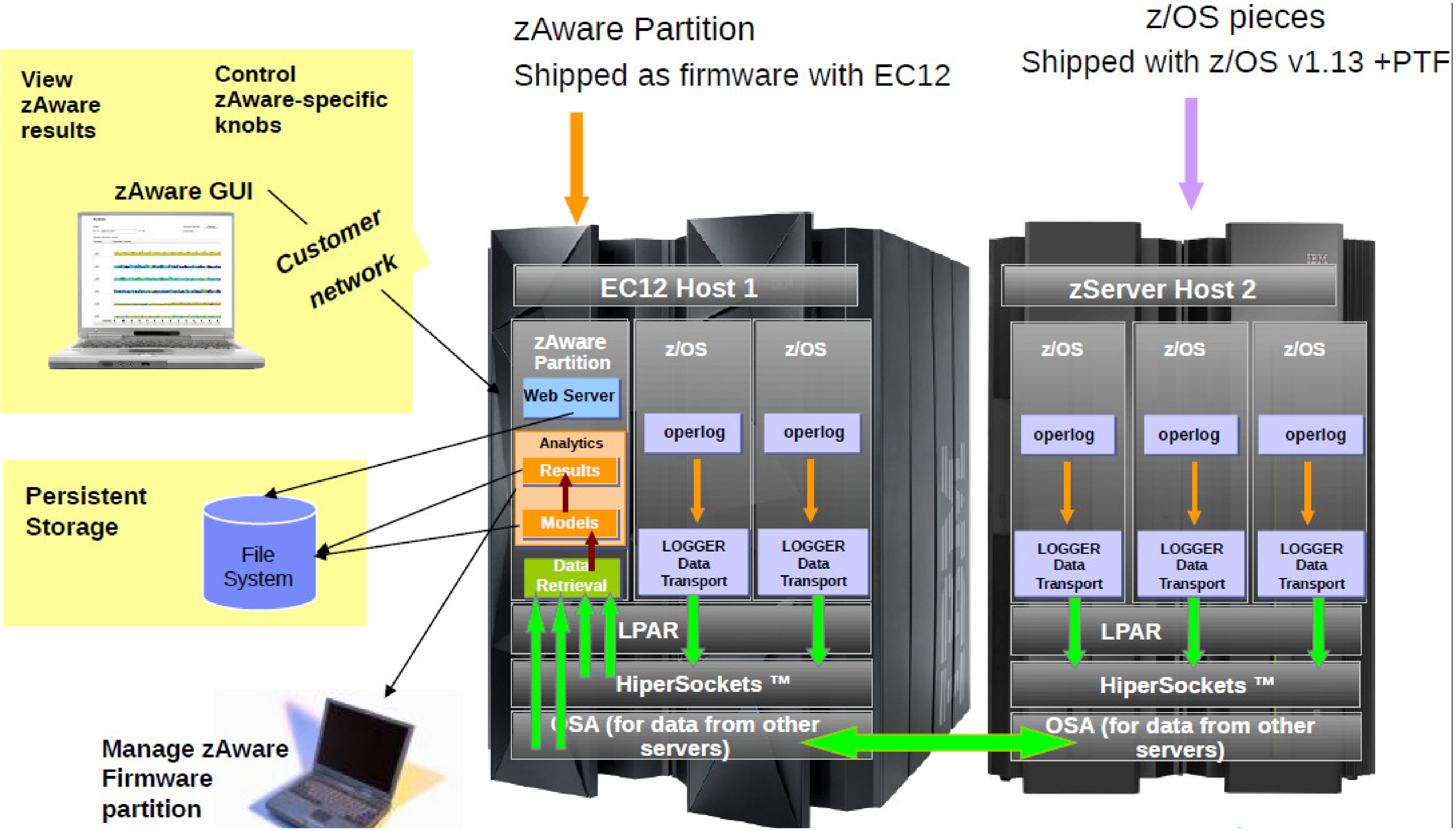
Some More Examples Of Typical Automated Performance Management Scenarios

- ***z/OS example*** - possible z/OS looping task
  - **Monitored symptoms** – high CPU loop index as measured by OMEGAMON >> WLM missing goals >> high overall system CPU usage
  - **Automation response** – adjust priority of problem task or if desired cancel the task
- ***DB2 example*** - DB2 object lock conflict
  - **Monitored symptoms** - long running SQL call >> high In-DB2 time >> longer thread elapsed time
  - **Automation response** - Increase priority of "owner" (as determined by automation) >> "Kill" problem thread
- ***IMS example*** - High IMS message region occupancy time
  - **Monitored symptoms** - IMS transactions queued >> longer IMS transaction scheduling time >> longer IMS response time >> lower IMS transaction processing rate
  - **Automation response** – automation starts additional message regions to handle workload >> issue IMS commands to adjust classes
- ***MQ example*** - Lower MQ message input rate >>
  - **Monitored symptoms** - Higher MQ message queue depth >> lower transaction processing rate >> longer CICS/IMS transaction response time
  - **Automation response** – issue calls to assess potential bottlenecks in CICS/IMS processing >> automation action based on results

Understanding Critical z/OS Messages
About zAware

- IBM zAware – IBM System z Advanced Workload Analysis Reporter
- Monitors z/OS OPERLOG including all messages written to z/OS console, including ISV and application generated messages
  - Early detection and focused diagnosis can help improve time to recovery
- Technology based on machine learning developed by IBM Research
  - Pattern recognition techniques look at the health of a system to pinpoint deviations from the 'norm'
  - High speed analytics facilitates the ability to consume large quantities of message logs
- Allow establishment of procedures to prevent reoccurrence
- IBM Red Book - http://www.redbooks.ibm.com/redbooks/pdfs/sg248070.pdf

Inside IBM zAware



zAware Partition

Shipped as firmware with EC12

z/OS pieces

Shipped with z/OS v1.13 +PTF

View zAware results

Control zAware-specific knobs

zAware GUI

Customer network

Persistent Storage

File System

Manage zAware Firmware partition

EC12 Host 1

zAware Partition

Web Server

Analytics

Results

Models

Data Retrieval

z/OS

operlog

LOGGER Data Transport

z/OS

operlog

LOGGER Data Transport

LPAR

HiperSockets ™

OSA (for data from other servers)

zServer Host 2

z/OS

operlog

LOGGER Data Transport

z/OS

operlog

LOGGER Data Transport

z/OS

operlog

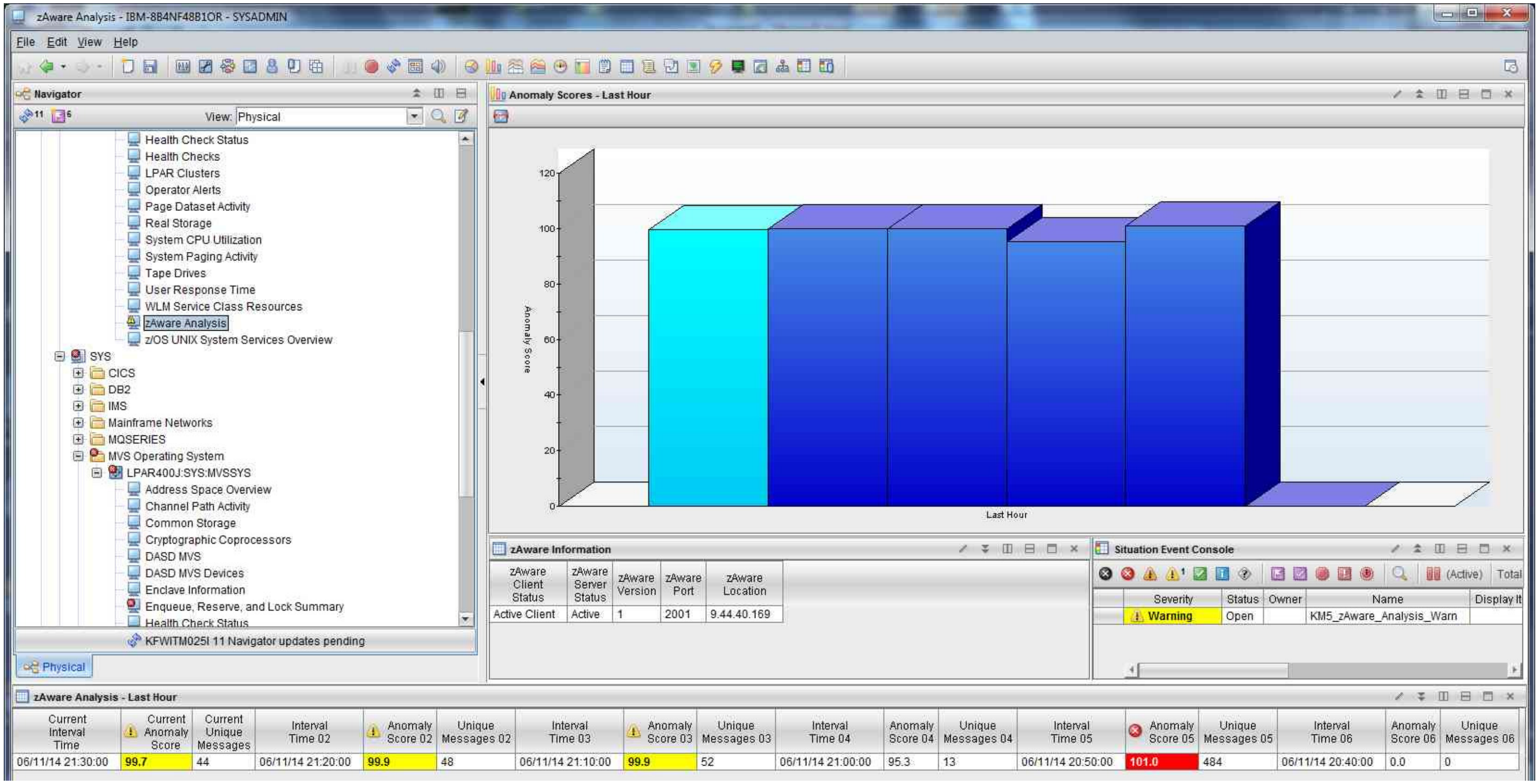
LOGGER Data Transport

LPAR

HiperSockets ™

OSA (for data from other servers)

zAware And Performance Monitoring



Navigator:

- Health Check Status
- Health Checks
- LPAR Clusters
- Operator Alerts
- Page Dataset Activity
- Real Storage
- System CPU Utilization
- System Paging Activity
- Tape Drives
- User Response Time
- WLM Service Class Resources
- zAware Analysis
- z/OS UNIX System Services Overview
- SYS
  - CICS
  - DB2
  - IMS
  - Mainframe Networks
  - MQSERIES
  - MVS Operating System
    - LPAR400J:SYS:MVSSYS
      - Address Space Overview
      - Channel Path Activity
      - Common Storage
      - Cryptographic Coprocessors
      - DASD MVS
      - DASD MVS Devices
      - Enclave Information
      - Enqueue, Reserve, and Lock Summary
      - Health Check Status

KFWITM025I 11 Navigator updates pending

zAware Information

zAware Client Status	zAware Server Status	zAware Version	zAware Port	zAware Location
Active Client	Active	1	2001	9.44.40.169

Situation Event Console

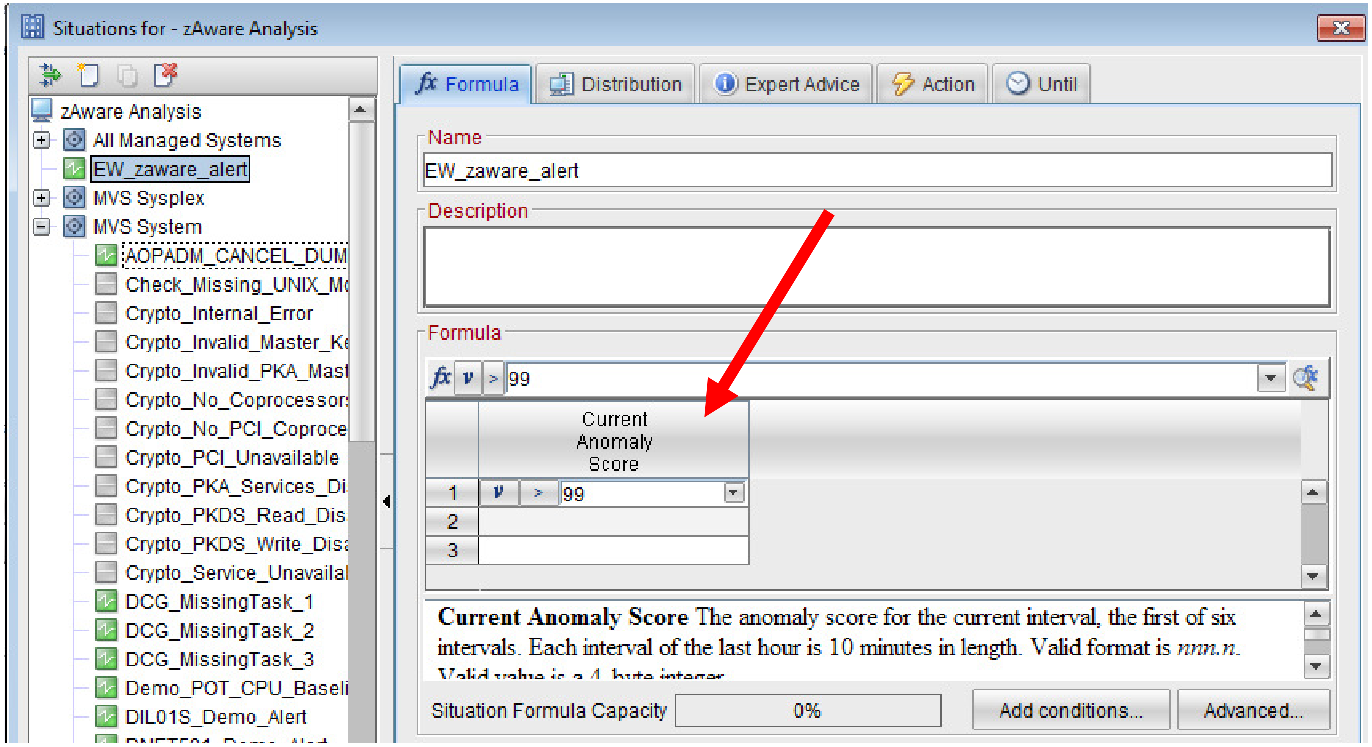
Severity	Status	Owner	Name	Display It
⚠ Warning	Open		KM5_zAware_Analysis_Warn	

zAware Analysis - Last Hour

Current Interval Time	⚠ Current Anomaly Score	Current Unique Messages	Interval Time 02	⚠ Anomaly Score 02	Unique Messages 02	Interval Time 03	⚠ Anomaly Score 03	Unique Messages 03	Interval Time 04	Anomaly Score 04	Unique Messages 04	Interval Time 05	⊗ Anomaly Score 05	Unique Messages 05	Interval Time 06	Anomaly Score 06	Unique Messages 06
06/11/14 21:30:00	99.7	44	06/11/14 21:20:00	99.9	48	06/11/14 21:10:00	99.9	52	06/11/14 21:00:00	95.3	13	06/11/14 20:50:00	101.0	484	06/11/14 20:40:00	0.0	0

- A new anomaly score is generated for each 10-minute period
- Anomaly scores from 99.6 to 100.9 are considered to be warning indicators
  - Indicate that there is message traffic in the monitored z/OS LPAR that is unusual
- Anomaly scores of 101 are considered critical and are even more important to investigate

Alerting On zAware Anomalies



- zAware Analysis
  - All Managed Systems
  - EW_zaware_alert
  - MVS Sysplex
  - MVS System
    - AOPADM_CANCEL_DUM
    - Check_Missing_UNIX_Mc
    - Crypto_Internal_Error
    - Crypto_Invalid_Master_Ke
    - Crypto_Invalid_PKA_Mast
    - Crypto_No_Coprocessor
    - Crypto_No_PCI_Coproce
    - Crypto_PCI_Unavailable
    - Crypto_PKA_Services_Di
    - Crypto_PKDS_Read_Dis
    - Crypto_PKDS_Write_Dis
    - Crypto_Service_Unavailal
    - DCG_MissingTask_1
    - DCG_MissingTask_2
    - DCG_MissingTask_3
    - Demo_POT_CPU_Baseli
    - DIL01S_Demo_Alert

Name: EW_zaware_alert

Description:

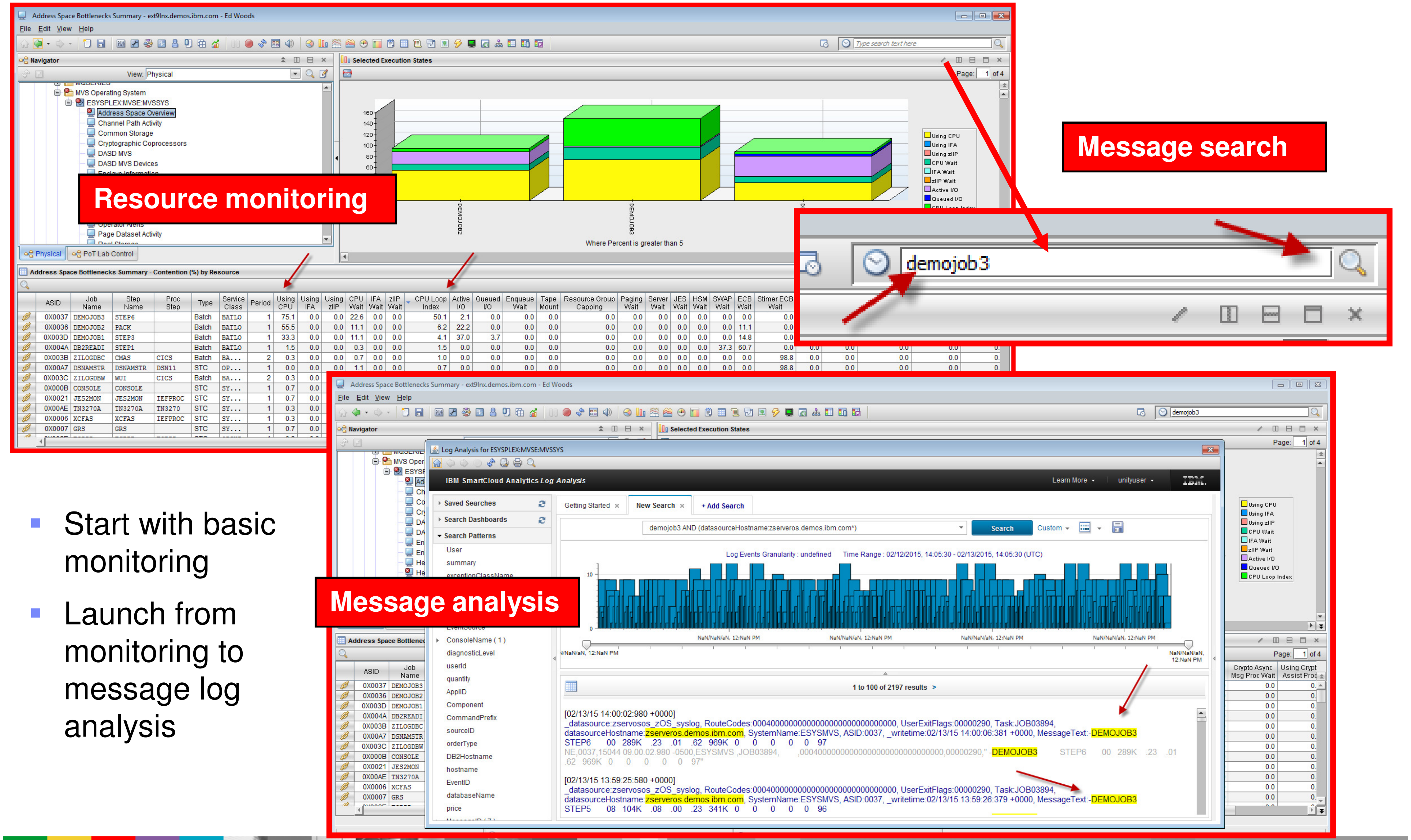
Formula: > 99

	Current Anomaly Score
1	> 99
2	
3	

Current Anomaly Score The anomaly score for the current interval, the first of six intervals. Each interval of the last hour is 10 minutes in length. Valid format is *nnn.n*. Valid value is a 4-byte integer.

- As with other situation alerts scenarios, an alert on a zAware anomaly may be used to drive notification or other analysis actions

Investigate Message Anomalies And Issues



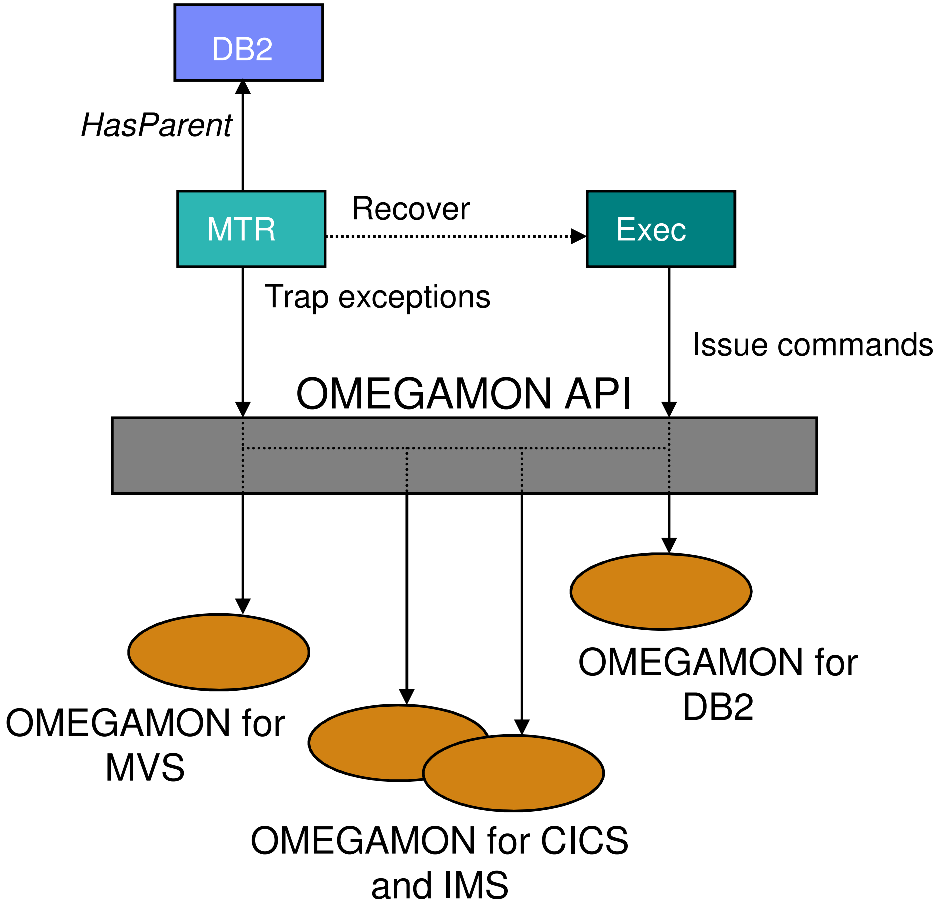
Address Space Bottlenecks Summary - Contention (%) by Resource

ASID	Job Name	Step Name	Proc Step	Type	Service Class	Period	Using CPU	Using IFA	Using zIIP	CPU Wait	IFA Wait	zIIP Wait	CPU Loop Index	Active I/O	Queued I/O	Enqueue Wait	Tape Mount	Resource Group Capping	Paging Wait	Server Wait	JES Wait	HSM Wait	SWAP Wait	ECB Wait	Stimer ECB Wait
0X0037	DEMOJOB3	STEP6		Batch	BATLO	1	75.1	0.0	0.0	22.6	0.0	0.0	50.1	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0X0036	DEMOJOB2	PACK		Batch	BATLO	1	55.5	0.0	0.0	11.1	0.0	0.0	6.2	22.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	11.1	0.0
0X003D	DEMOJOB1	STEP3		Batch	BATLO	1	33.3	0.0	0.0	11.1	0.0	0.0	4.1	37.0	3.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	14.8	0.0
0X004A	DB2READI	STEP1		Batch	BATLO	1	1.5	0.0	0.0	0.3	0.0	0.0	1.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	37.3	60.7	0.0
0X003B	ZILOGDBC	CMAS	CICS	Batch	BA...	2	0.3	0.0	0.0	0.7	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	98.8
0X00A7	DSNAMSTR	DSNAMSTR	DSN11	STC	OP...	1	0.0	0.0	0.0	1.1	0.0	0.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	98.8
0X003C	ZILOGDBW	WUI	CICS	Batch	BA...	2	0.3	0.0																	
0X000B	CONSOLE	CONSOLE		STC	SY...	1	0.7	0.0																	
0X0021	JES2MON	JES2MON	IEFPROC	STC	SY...	1	0.7	0.0																	
0X00AE	TN3270A	TN3270A	TN3270	STC	SY...	1	0.3	0.0																	
0X0006	XCFAS	XCFAS	IEFPROC	STC	SY...	1	0.3	0.0																	
0X0007	GRS	GRS		STC	SY...	1	0.7	0.0																	

```
[02/13/15 14:00:02:980 +0000]
_datasource:zservosos_zOS_syslog, RouteCodes:00040000000000000000000000000000, UserExitFlags:00000290, Task:JOB03894,
datasourceHostname:zserveros.demos.ibm.com, SystemName:ESYSMVS, ASID:0037, _writetime:02/13/15 14:00:06:381 +0000, MessageText:-DEMOJOB3
STEP6     00  289K  .23   .01   .62  969K  0    0    0    0    0   97
NE,0037,15044 09.00.02.980 -0500,ESYSMVS ,JOB03894,      ,00040000000000000000000000000000,00000290," -DEMOJOB3      STEP6     00  289K  .23   .01
.62  969K  0    0    0    0    0   97"

[02/13/15 13:59:25:580 +0000]
_datasource:zservosos_zOS_syslog, RouteCodes:00040000000000000000000000000000, UserExitFlags:00000290, Task:JOB03894,
datasourceHostname:zserveros.demos.ibm.com, SystemName:ESYSMVS, ASID:0037, _writetime:02/13/15 13:59:26:379 +0000, MessageText:-DEMOJOB3
STEP5     08  104K  .08   .00   .23  341K  0    0    0    0    0   96
```

- Start with basic monitoring
- Launch from monitoring to message log analysis

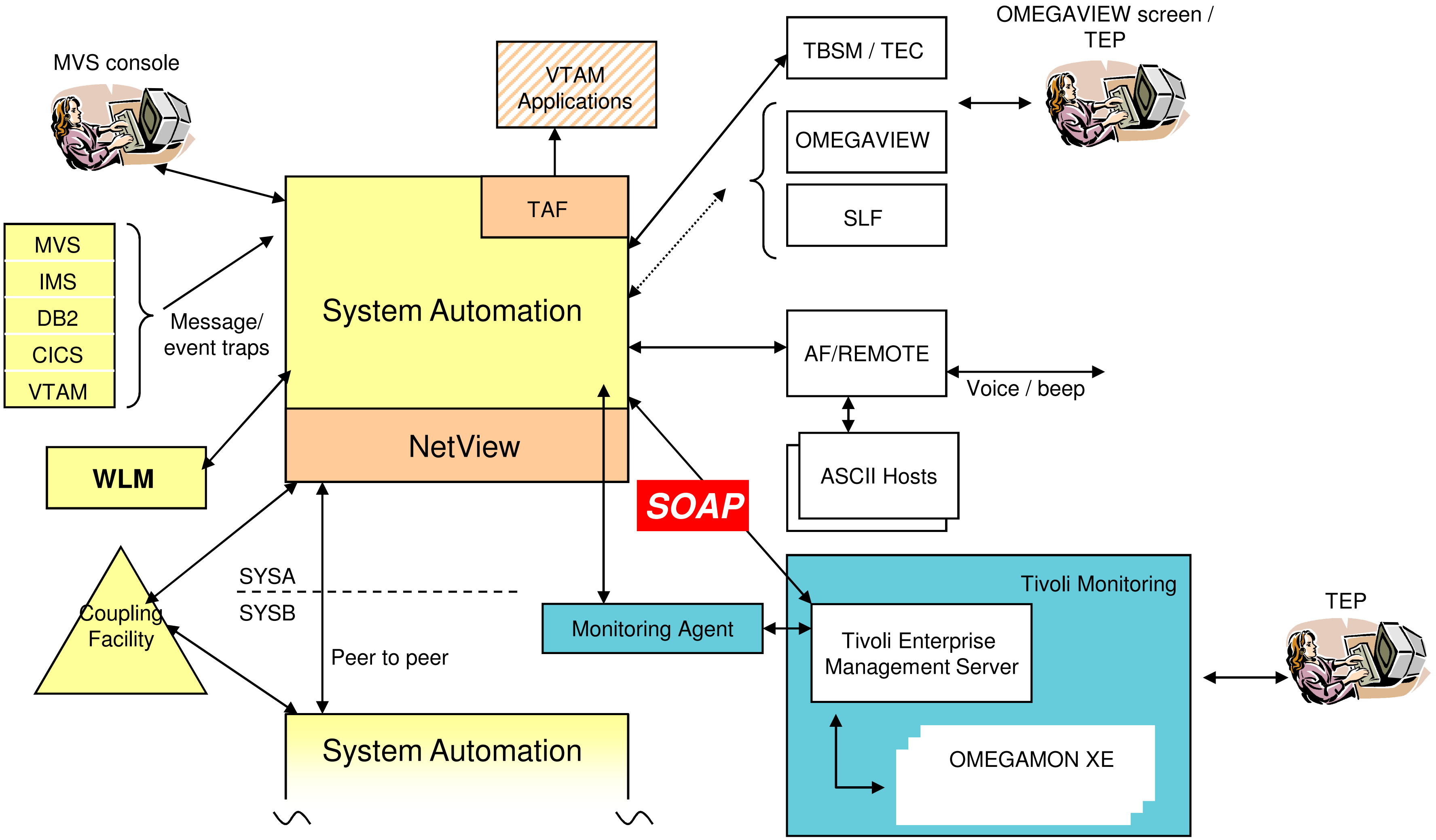
Automation Integrated With Monitoring Example – The Need For Bi-directional Interfaces



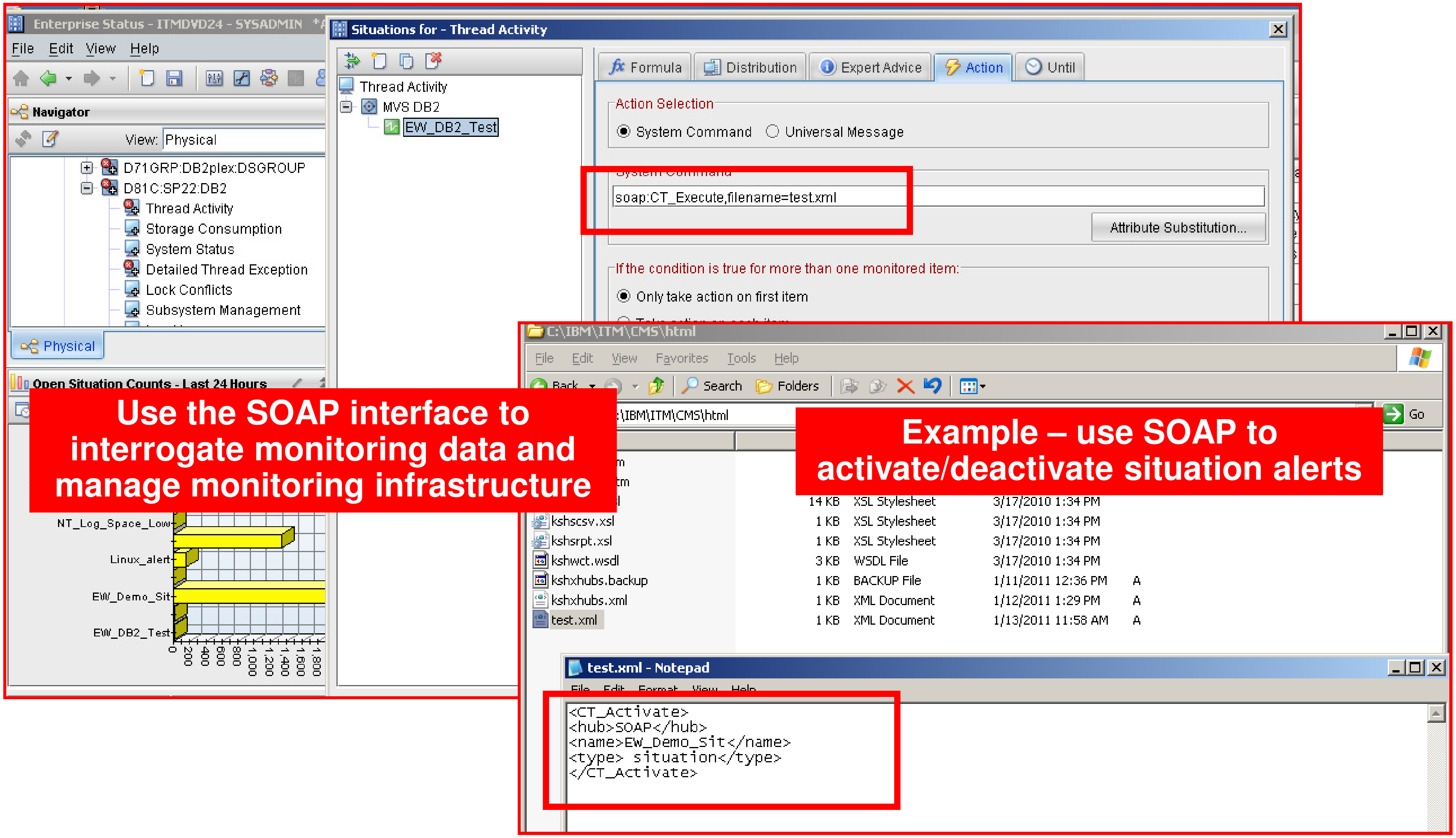
- **Use performance and availability information for Automated Performance Management**
  - **More metrics, more accurate decisions**
  - **Sources: MVS, DB2, CICS, IMS, Network, Webpshere, Websphere MQ, Storage monitoring**
- **Provides APIs to communicate with OMEGAMON monitors to**
  - **Monitor OMEGAMON exceptions**
  - **Monitor/manage situation status**
  - **SOAP interface enables detailed performance data interface to SA**
- **Provides exception monitor based on the Monitor Resource concept**
  - **Monitors „interesting“ set of exceptions**
  - **Sets application health state based on existence of such exceptions**
  - **Provides means to react and resolve exceptional conditions**

Interface means any metric captured by OMEGAMON may be analyzed via automation

SA / OMEGAMON Integration
SOAP Interface Enables Detailed Analysis



OMEGAMON, IBM System Automation And The Tivoli Enterprise Portal Provides SOAP Interface



Action Selection

● System Command ○ Universal Message

System Command

soap:CT_Execute,filename=test.xml

If the condition is true for more than one monitored item:

● Only take action on first item

○ Take action on each item

:\IBM\ITM\CMS\html

l	14 KB	XSL Stylesheet	3/17/2010 1:34 PM	
kshscsv.xsl	1 KB	XSL Stylesheet	3/17/2010 1:34 PM	
kshsrpt.xsl	1 KB	XSL Stylesheet	3/17/2010 1:34 PM	
kshwct.wsdl	3 KB	WSDL File	3/17/2010 1:34 PM	
kshxhubs.backup	1 KB	BACKUP File	1/11/2011 12:36 PM	A
kshxhubs.xml	1 KB	XML Document	1/12/2011 1:29 PM	A
test.xml	1 KB	XML Document	1/13/2011 11:58 AM	A

```
<CT_Activate>
<hub>SOAP</hub>
<name>EW_Demo_Sit</name>
<type> situation</type>
</CT_Activate>
```

Use the SOAP interface to interrogate monitoring data and manage monitoring infrastructure

Example – use SOAP to activate/deactivate situation alerts

Performance Automation Integration Within A Common Dashboard Or Portal

- The Portal provides manual commands and corrections
  - ‘Take Action’ provides for manual command capability
  - Commands may be predefined
- The Portal enables automated commands and corrections
  - Implement machine speed corrective actions, issue alerts, and allow for later human intervention
  - Use for automated commands for dynamic subsystem management and ‘tweaks’ as the workload and system changes
  - Two core types of automated actions
    - ***Situations*** - Use for simple “fire and forget” type of scenarios
    - ***Policies*** – Use for more sophisticated performance automation scenarios

About Situations And Policies

- Alerts (aka Situations) are the building blocks of systems management logic
  - Situations may be used to highlight performance and availability problems within key operating systems, subsystems, and mission critical resources
- Policies extend concepts established with situations and add additional functionality
  - Situations remain the essential starting point
  - Policies add additional function and flexibility
- Start with the basic building blocks and grow from there

Additional Situation Considerations And Recommendations

- Use the Product Provided Situations as examples or templates
  - Customization to user-created situations
- When creating and deploying a set of situations consider
  - The number of situations being deployed
  - The number of managed systems (i.e. z/OS LPARs and CICS tasks)
  - Refresh frequency of the situations
- Consider carefully the number of required situations
  - Use boolean logic to reduce the number of needed situations
  - Do not automatically make a warning alert to go with each critical alert
    - Create a warning if it will allow time to address an issue before going critical
  - Use managed system lists to send the right situations to the right managed systems
- Be aware of the situation refresh rates
  - Multiple situations on the same table with the same refresh rate may be optimized by the infrastructure
  - Potential to reduce monitoring overhead if done appropriately

Policies And System Automation Recommendations And Rules Of Thumb

- Policies are not a substitute for System Automation and REXX command script capabilities
  - Policies work well as an extension of situation capabilities
  - Policies work well to manage start/stop of situation logic
  - Policies work well to issue multiple actions and “feed” other tools
- IBM System Automation
  - Use for full function automation logic and routines
    - REXX exec script capabilities
  - Use for more complex logic and actions
  - Exploit the ability of the SOAP interface to pull in key performance metrics from OMEGAMON

Roadmap
Automated Performance Management

- Use a building block approach
- ***Situations*** - Start with identification and definition of situation alerts
  - Meaningful alerts that represent true potential issues
  - Use the analysis to identify critical monitoring metrics
- ***Policies*** – Use policies where appropriate
  - Situation management and correlation
  - Issuing commands for basic performance/availability issues
- ***Visualization*** – Define useful Tivoli Portal views
  - Customize screens in the Portal for specific audiences
    - Operations, applications, management
- ***System Automation*** – exploit the power of integration
  - Define example performance automation management scenarios
  - Leverage the process as a template for additional scenarios

Summary

- Exploit your Monitoring and Automation suite provides powerful automation capabilities in multiple core technologies
  - Automation console management
  - Resource monitoring
  - Network monitoring and Event management
- Automated Performance Management leverages the intrinsic integration capabilities of the various technologies
  - Automation integration with monitoring
  - Integrated monitoring and management (including cross platform)
- Leverage Automated Performance Management to improve problem isolation and MTTR
  - Understand the unique capabilities of integrated monitoring and automation
  - Use a building block approach to grow management logic over time

Thank You!!