IBM Software Group

Performance Automation Best Practices Using Policies and the Tivoli Enterprise Portal

Ed Woods

Consulting IT Specialist

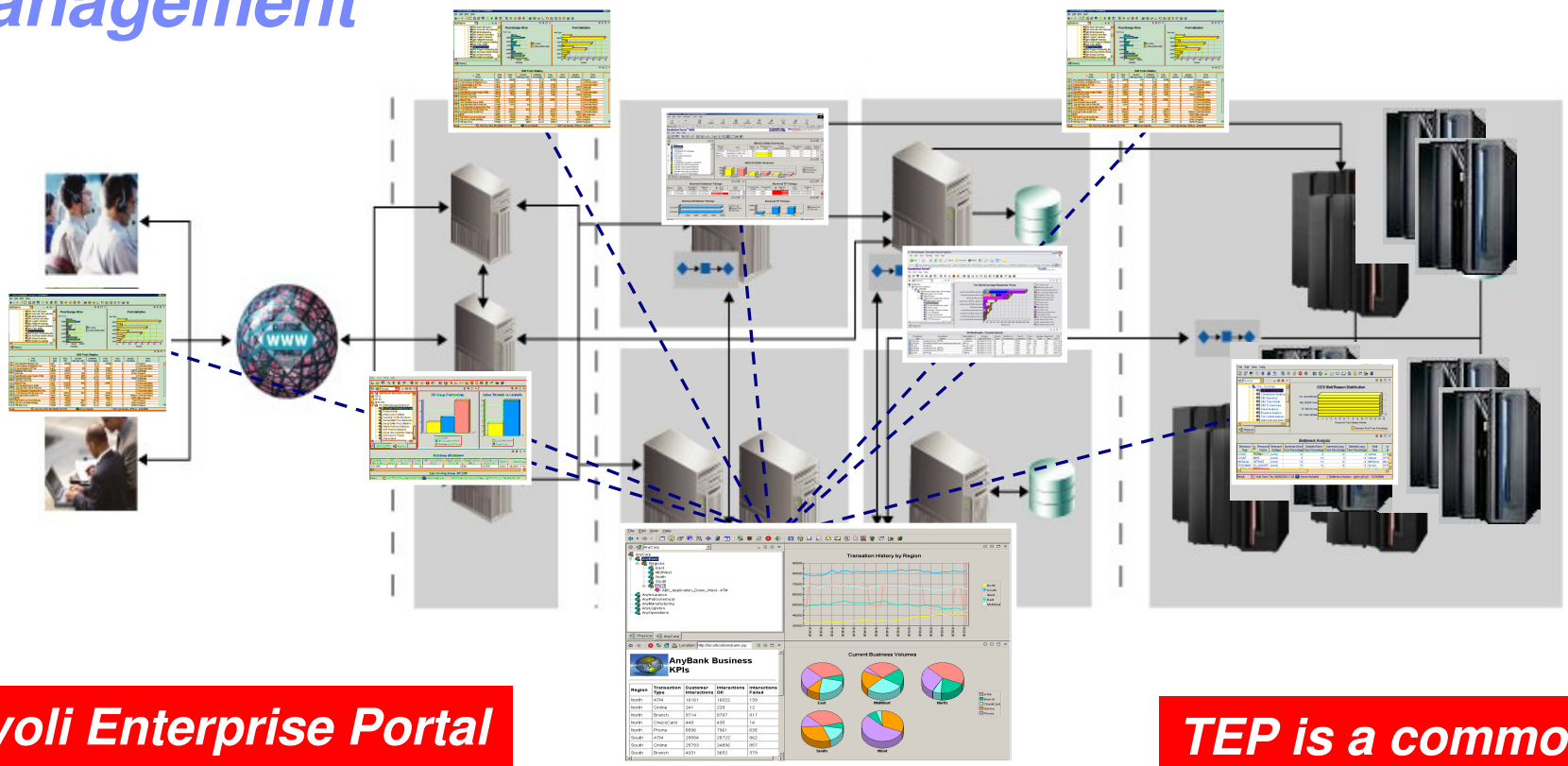
Tivoli software

@business on demand.

Agenda

- The Tivoli Enterprise Portal And Integrated Performance And Availability Automation
- About Situations And Policies
- Situation Review
- Policies – Definition And Deployment
- Policy Usage Recommendations And Best Practices
- Summary And Questions

Tivoli Enterprise Portal (The TEP) Integrated Performance, Availability, And Systems Management



Tivoli Enterprise Portal (TEP)

Tivoli Enterprise Portal enables integrated alert and automation capabilities

TEP is a common user interface for a variety of Tivoli solutions

Tivoli Enterprise Portal
Integrated Performance And Availability Automation

- The TEP provides manual commands and corrections
  - 'Take Action' provides for manual command capability
  - Commands may be predefined
- The TEP 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
  - Situations - Use for simple "fire and forget" type of scenarios
  - Policies – Use for more sophisticated performance automation scenarios

Note - Policy automation requires OMEGAMON Dashboard Edition (DE) which is a separately licensable item on z/OS

Tivoli Enterprise Portal
Integrated Command And Automation Options

- The TEP provides multiple command options
  - Manual ‘Take Action’, Situations, Policies
- Take Action provides for manual command capability
  - Commands may be predefined
- Situations are the basic building blocks for alerts and notification
  - Situations drive alerts and notification
  - Situations offer automated reflex action command function
- Policies allow for multiple step commands, checks, and automated actions
  - Policies consist of a combination of situations, commands, and other checks
  - Policies provide added flexibility and power

About Situations And Policies

- Situations are the building blocks of systems management logic in the Tivoli Enterprise Portal (TEP)
  - Situations may be used to highlight performance and availability problems within key operating systems, subsystems, and mission critical resources
  - Situation logic may be distributed to the agent (IRA architecture)
    - Situations typically run at the level of the agent (TEMA)
- Policies extend concepts established with situations and add additional functionality to the TEP
  - Situations remain the essential starting point
  - Policies add additional function and flexibility
  - Policies run within the TEMS infrastructure

Understanding The Capabilities Of Situations Powerful, Flexible Alerts And Automation

- OMEGAMON XE situation capabilities allow for more intelligent alerts that integrate and correlate status and information
- Situations may incorporate Boolean logic
- Situations may be correlated with other situations
- Situations may in turn drive automated corrections
- Situations are the essential starting point for policy automation

Situations – The Starting Point
Highlight Performance And Availability Issues

Welcome DNET581
Tivoli Enterprise Portal
Log out

File Edit View Help

View: Physical

Situation Event Console

Total Events: 38 | Item Filter: Enterprise

Severity	Status	Owner	Situation Name	Display Item	Source
Informatio...	Open		Kah_Oper_Requests_Exist_Info		DEMOPLX:DEMOPLX:S
Informatio...	Open		Kah_Mtr_Health_Status_Info	DEMO_CPU	DEMOPLX:DEMOPLX:S
Warning	Open		ZVM_Avail_Mean2G_Low		zdemolx.demopkg.ibm
Warning	Open		N3T_Conn_Rnd_Trip_Variance		TCPIP:MVSA
Warning	Open		N3V_Pct_ECSA_Allocated_Stg		VTAM:MVSA
Warning	Open		Kah_Resource_Health_Warn	DEMO_CBJ	DEMOPLX:DEMOPLX:S
					DEMOPLX:DEMOPLX:S
					DEMOPLX:MVS:SYSPL
					DEMOPLX:DEMOPLX:S
					DEMOPLX:DEMOPLX:S
					TCPIP:MVSA

DB2
DB2S:MVSA:DB2
DSNA:MVSA:DB2
Thread Activity
System Status
Detailed Thread Exception

CRITICAL
EW_Thread_Alert DSNA:MVSA:DB2 07/02/07 08:49:51

KFWTM101I Select workspace link button to view situation event results.

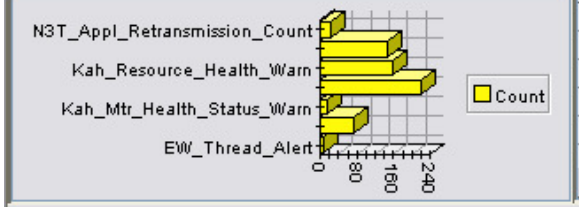
Physical

Click to see alert detail

Flyover pop-up shows the name of the 'situation' alert

Acknowledged Events
Page Log

Status	Name	Display Item	Origin Node	Global Time
Open	Kah_Resource_Health_Crit	DEMO_SRVR	DEMOPLX:DEMOPLX:SA	07/02/07 13
Open	Kah_Resource_Health_Crit	SYSPLEX	DEMOPLX:DEMOPLX:SA	07/02/07 13
Open	Kah_Resource_Health_Crit	DEMOMN2	DEMOPLX:DEMOPLX:SA	07/02/07 13
Open	Kah_Rsrc_Not_Satisfactory_Crit	DEMO_SRVR	DEMOPLX:DEMOPLX:SA	07/02/07 13
Open	Kah_Rsrc_Not_Satisfactory_Crit	DEMOMN2	DEMOPLX:DEMOPLX:SA	07/02/07 13
Open	Kah_Mtr_Health_Status_Crit	DEMOMN2	DEMOPLX:DEMOPLX:SA	07/02/07 13



Situations - A Basic Example
Alert On DB2 Threads With More Than 'n' Getpages

Situations for - Detailed Thread Exception

Detailed Thread Exception
EW_Threat
MVS DB2
KDP_V
KDP_V

Create New...
Create Another...
Start Situation
Stop Situation
Delete Situation
Associate
Dissociate

Formula | Distribution | Expert Advice | Action | Until

Description

Formula

	Getpage Count
1	> 1000
2	
3	

Click inside a cell of the formula editor to see a description of the attribute for that column and to compose the expression.

Add a condition by clicking **Add conditions** and selecting the situations to embed or attributes you want to include.

Situation Formula Capacity 5% | Add conditions... | Advanced...

Sampling interval: 0 / 0 : 1 : 30 (ddd hh mm ss)

Sound: Enable critical.wav | Play | Edit...

State: Critical | Run at startup

OK | Cancel | Apply | Help

Start/stop situation

Distribution tab to specify where situation runs.
Expert advice is customizable.
Action tab to execute command.

Specify alert criteria.
This may include one or multiple attribute criteria.

Specify sampling interval

Specify severity and whether to run at Omegamon startup

Situations
'Action' To Perform Commands And Corrections

Situations for - Detailed Thread Exception

Detailed Thread Exception
MVS DB2
EW_Thread_Alert
KDP_WTRE_Critical
KDP_WTRE_Warning

Formula | Distribution | Expert Advice | Action | Until

Action Selection
System Command | Universal Message

System Command
LOG 'DB2 Thread Alert Message - &{DB2_Thread_Exceptions.Plan_Name} &{DB2_Thread_Exceptions.Authorization_

Attribute Substitution...

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 Action at the Managing System (TEMS)

If the condition stays true over multiple intervals:
Don't take action twice in a row (wait until situation goes false then true again)
Take action in each interval

Where command is executed

Attribute substitution in the command line

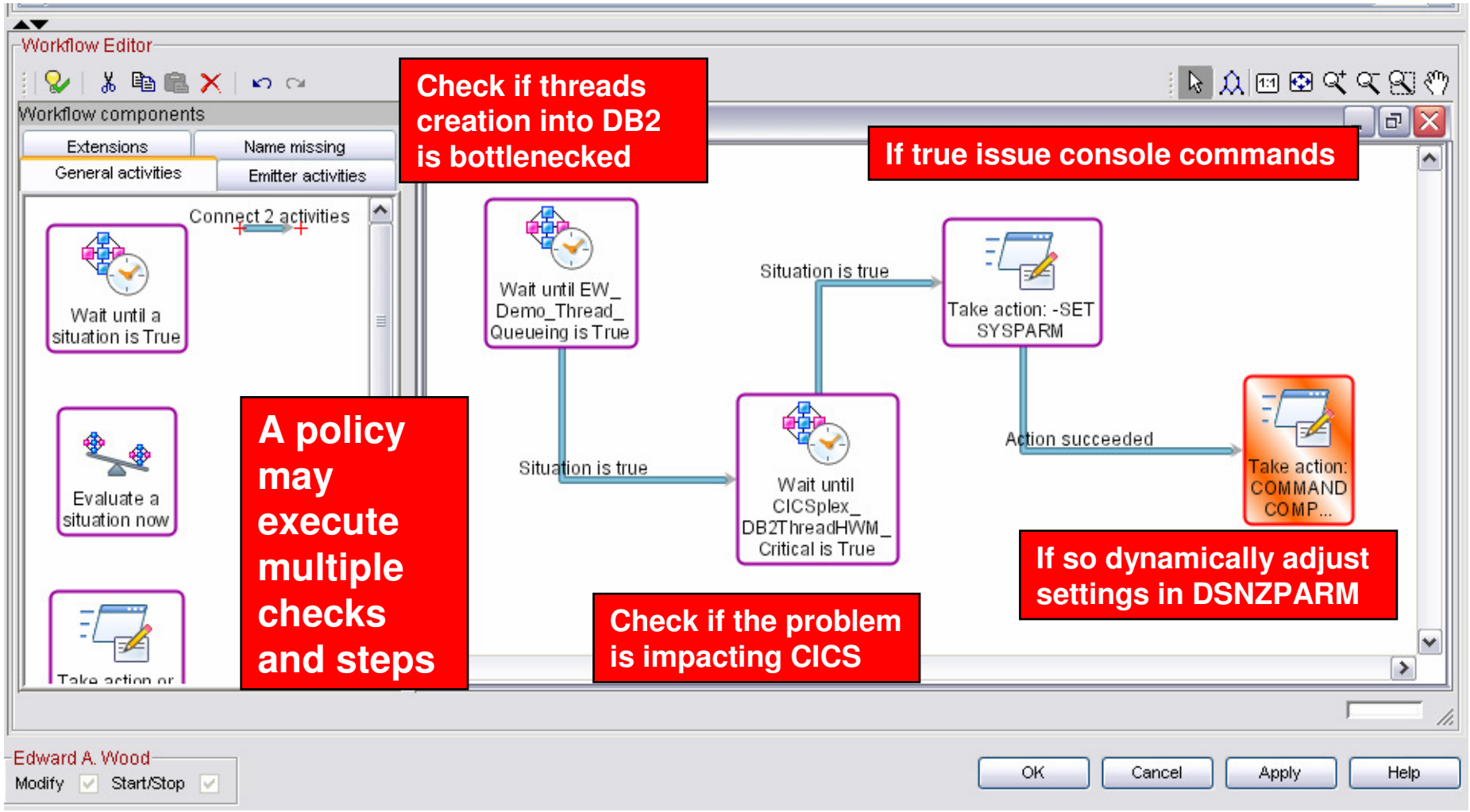
System commands may be executed when the situation is true

Examples of actions include:
- **DB2 thread kill command**
- **Issuing messages to the console**
- **Any valid z/OS console command**
- **Issuing commands to drive notification**

Situations
General Recommendations And Rules Of Thumb

- Make situations Meaningful, Actionable, and Useful
- Meaningful situations
  - Situation naming is flexible – make the names understandable
  - Adopt a situation naming convention
    - Makes it easier to identify customer created versus product provided situations
- Actionable situations
  - 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 situations
  - Eliminate phony alert indicators – tune out the noise
  - If an alert situation fires it should indicate an actual issue
    - An alert, an owner, and a consequence

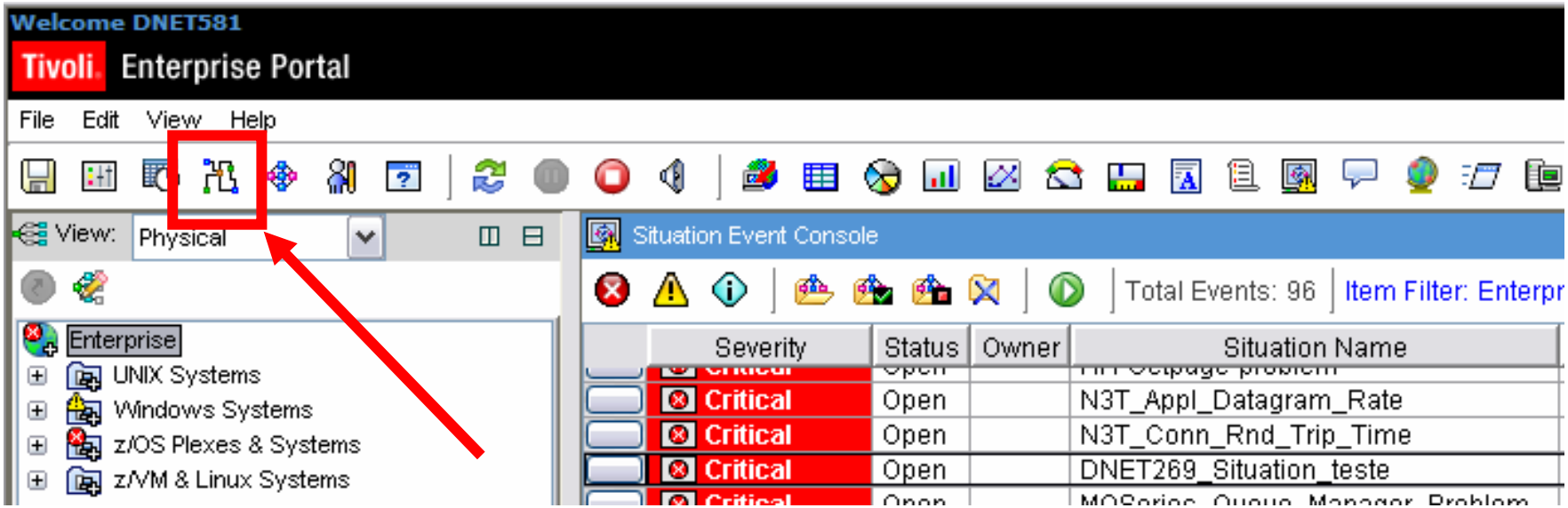
Policies Expand The Concept Of Integrated Performance Automation



Policies And Performance Automation

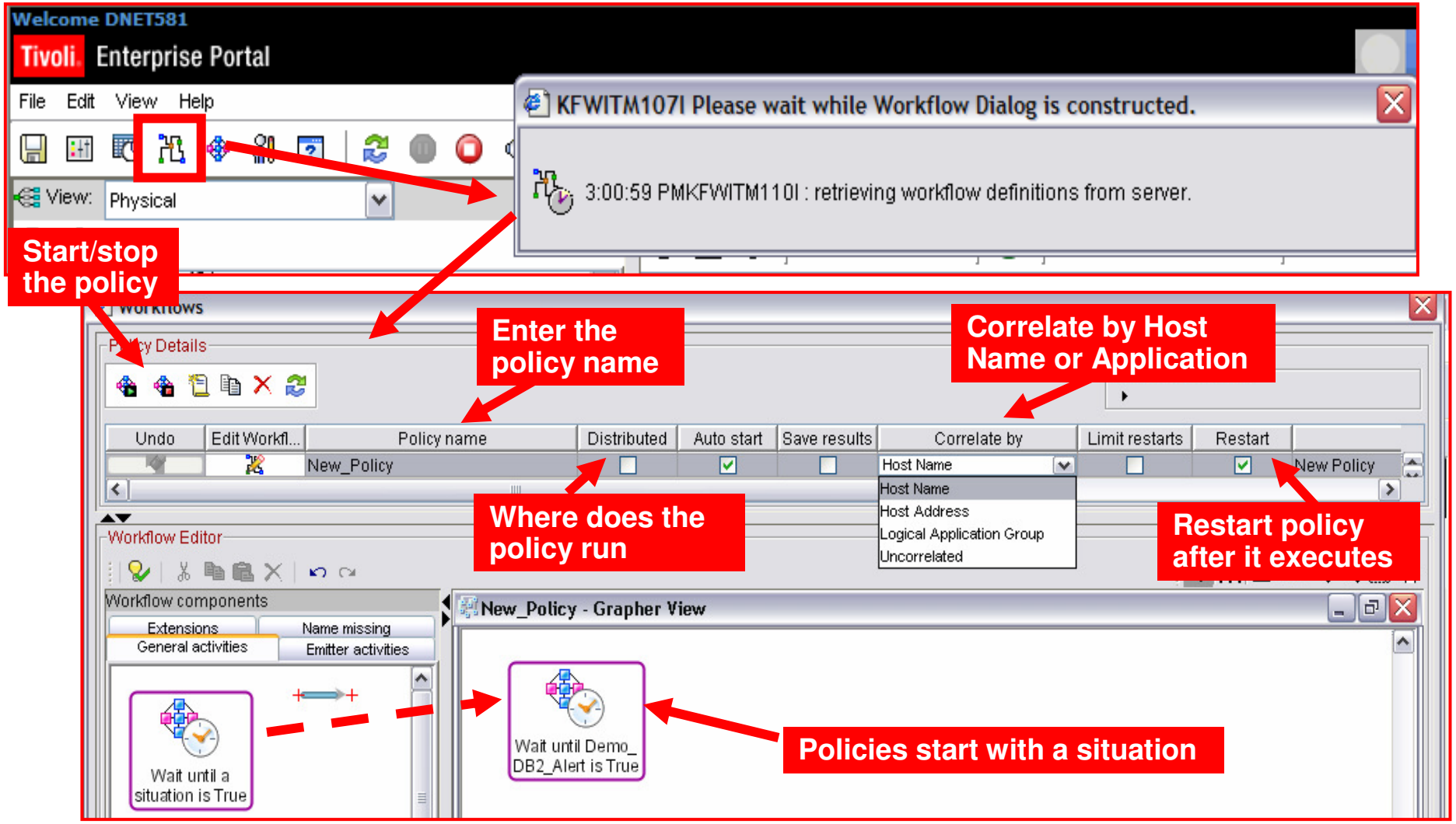
- Policies allow for more sophisticated performance automation scenarios
  - Multiple situation checks, multiple commands
- Policies allow for the implementation of machine speed corrective actions and issuing alerts
- Potential uses of policies include
  - Basic alert correlation with corrective actions
  - Forwarding of alerts to other alert managers
  - Use as a mechanism to feed other automation technologies
  - Dynamic management of monitoring infrastructure

Tivoli Enterprise Portal – Policy Support Requirements And Pre-Requisites



- Policy support requires OMEGAMON DE (Dashboard Edition) enablement
- OMEGAMON DE is included for ITM 6.1 distributed monitoring customers
- OMEGAMON DE is a separate component with z/OS monitoring

To Launch The Policy Editor



Policy Editor – Actions And Options

Workflows

Policy Details

Compatibility levels

Undo	Edit Workfl...	Policy name	Distributed
		dnet581_demo_basic_pol	✓

Workflow Editor

Workflow components

General activities | Emitter activities | Extensions | Name missing

Wait until a situation is True

Evaluate a situation now

Take action or Write message

Make a choice

Suspend execution

Start/Stop a policy

Start/Stop a situation

Edward A. Wood

Modify ✓ Start/Stop ✓

Evaluate situations
Take action
Start/stop situations & policies

Workflow Editor

Workflow components

General activities | Emitter activities | Extensions | Name missing

TEC_Event

ITO_Event

NetView_Event

SNMP Event

Emit alerts to TEC or SNMP

Workflow Editor

Workflow components

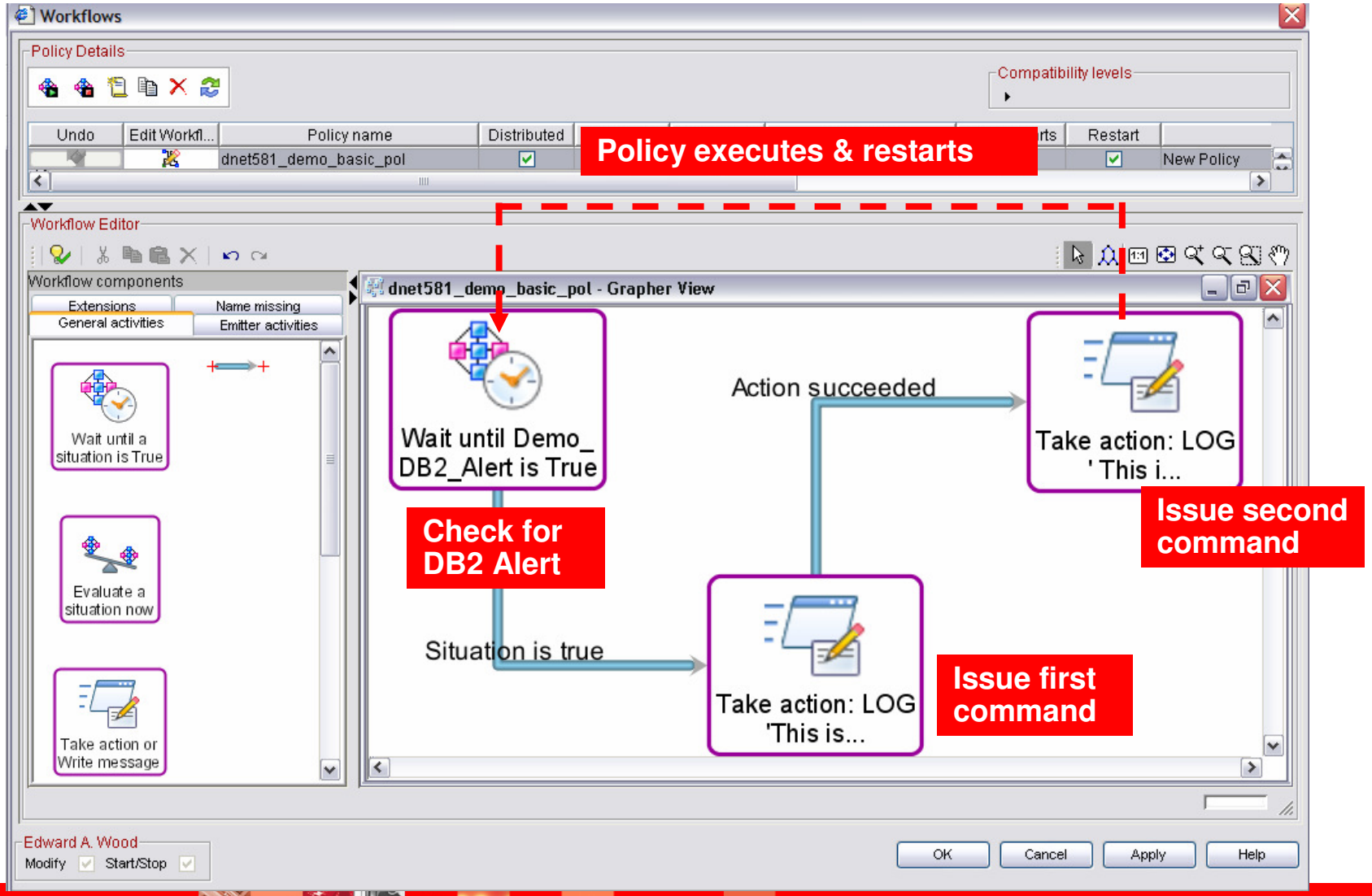
General activities | Emitter activities | Extensions | Name missing

Wait until a situation is False

Wait until situation is false to restart the policy

The policy editor provides multiple options to issue commands, emit alert info, control the flow of policy execution, and start/stop situations and policies.

Basic Policy - Example Scenario
Have A Situation Trigger Multiple Commands



Take Action Options Within A Policy

Workflow Editor

Workflow components

Extensions | Name missing | General activities | Emitter activities

Wait until a situation is True

Evaluate a situation now

dnet581_demo_basic_pol - Grapher View

Wait until Demo_DB2_Alert is True

Situation is true

Action Settings

Action Selection

Universal Message | System Command

System Command

LOG 'This is a test message - DB2 message &WaitOnSituation1:DB2_Thread_Exceptions.Plan_Name'

Attribute...

Fewer options

If multiple data items are returned

Only take action on the first item

Take action on each item from activity: WaitOnSituation1

Continue processing rows on failure

Where should the action be executed

Execute the action at the agent that returned data for activity: WaitOnSituation1

Execute the action at the TEMS

Execute the action at .OMEGLNX2:LZ

OK | Cancel | Help

Attribute substitution in the command line

Take the action at the agent
Which agent?
Where is the agent?

Take the action at the TEMS
Which TEMS?
Where is the TEMS?

Note - Where the action gets executed dictates the appropriate type of action

Distribution Controls Where Policy Is Executed And How Situations Are Evaluated

Policy Details

Click 'Distributed' to specify policy distribution. Controls how/where the policy runs.

Start/stop the policy

Change Policy Distribution

Assigned: DSNC:MVSA:DB2

Available Managed Systems: .OMEGLNX2:LZ, BWF0:MVSA:MQESA, BWF1:MVSA:MQESA, CNM16, CXEGDSSR:MVSC:STORAGE, CXEGDSST:MVSA:STORAGE, CXEGN3:MVSA:KN3AGENT, CXEGN3:MVSB:KN3AGENT

Available Managed System Lists: *ALL_CMS, *CPIRA_MGR, *CUSTOM_UAGENT00

Specify a managed system for the situation – in this example DSNC:MVSA:DB2

Remember policies run in a TEMS

At policy start the user will be prompted to select a TEMS

To See The Status Of Policy Distribution To A Managed System

Welcome DNET581
Tivoli Enterprise Portal
Log out
File Edit View Help
View: Physical
Situation Event Console
Manage Policy at Managed System: DSNC:MVSA:DB2

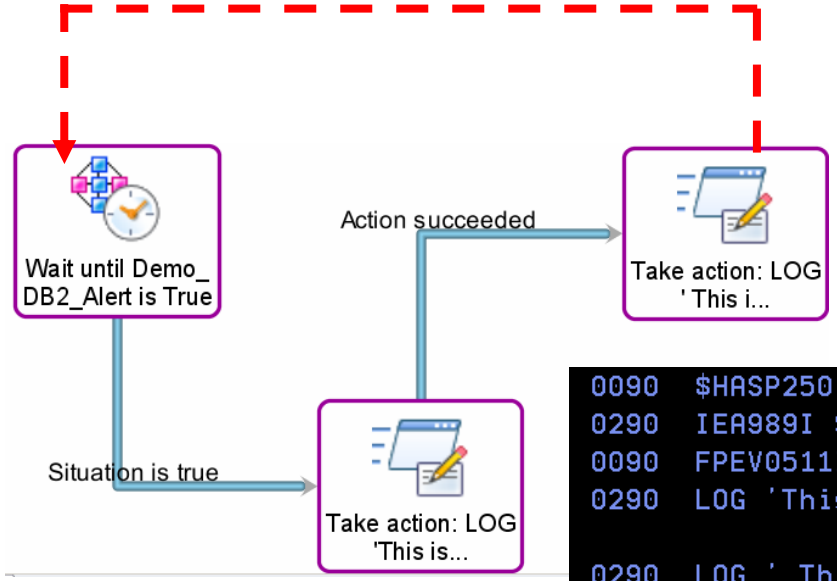
Name	Version	Auto Start	Distributed	Limit restar...	Restart	Description
dnet581_demo_basic_pol	04.00.00		✓		✓	New Policy
dnet581_demo_pol2	03.50.00		✓		✓	New Policy
DNET581_DEMO_POLICY	03.50.00		✓		✓	New Policy

Close | Select All | Deselect All | Help

Situations...
Manage Situations
Manage Policies
Print Preview...
Print...
Properties...

Hub Time: Thu, 01/10/2008 09:06 AM | Server Available | Enterprise - hqdnt2.demopkg.ibm.com - DNET581

Policy Command Execution



In the example the policy will:

Check the situation status
Execute the first command
Execute the second command
Restart

```
0090  $HASP250 DNET145 PURGED -- (JOB KEY WAS C1C5C854)
0290  IEA989I SLIP TRAP ID=X33E MATCHED.  JOBNAME
0090  FPEV0511I  DSNB HISTORY DATA SET WRAPPED, 4272 INTERVALS STORED
0290  LOG 'This is a test message - DB2 message ADHPLAN3'

0290  LOG ' This is a second test message'

0290
0290  IEA989I SLIP TRAP ID=X33E MATCHED.  JOBNAME=
0290  LOGON
0290  LOG 'This is a test message - DB2 message ADHPLAN3'

0290  LOG ' This is a second test message'

0281  $HASP100 DNET581  ON TSOINRDR
0090  $HASP373 DNET581  STARTED
0090  IEF125I DNET581 - LOGGED ON - TIME=10.06.27
```

First command

Second command

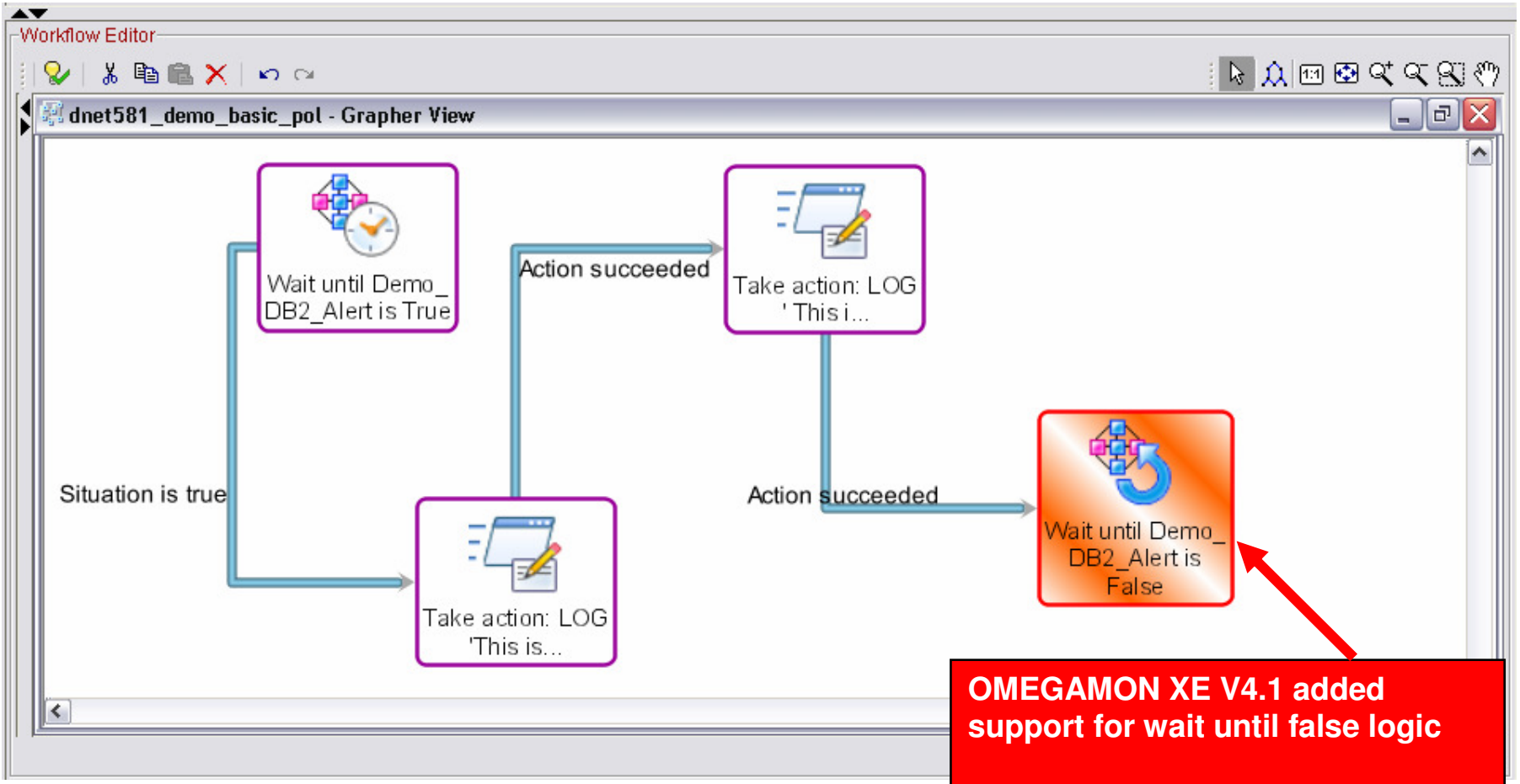
Restart

First command

Second command

Note – The interval of the situation will have an impact on the duration of the policy

Wait Until False Logic Expands The Flexibility Of Policies



OMEGAMON XE V4.1 added support for wait until false logic

Allows additional logic and flexibility in policy support

Policy Example
Multiple Situations, Multiple Commands

Policy Details

Compatibility levels

Undo	Edit Workfl...	Policy name	Distributed	Auto start	Save results	Correlate by
		DNET581_DEMO_POLICY	✓			Host Name

Host Name
Host Address
Logical Application Group
Uncorrelated

Correlate by host name

Workflow Editor

Workflow components

Extensions | Name missing
General activities | Emitter activities

Wait until a situation is True

DNET581_DEMO_POLICY - Grapher View

Wait until Demo_DB2_Alert is True

Check DB2 alert

Action succeeded

Wait until MQSeries_MQ_Channel_Stopped is True

Check MQ alert

Situation is true

Take action: LOG TEST ME...

Issue command

Situation is true

Take action: LOG TEST ME...

Issue command

Note – The DB2 alert and the MQ alert are independent events. The same would apply for two situations of the same agent type or managed system.

Distribution Of The Policy

Policy Details

Compatibility levels

Undo	Edit Workfl...	Policy name	Distributed	Auto start	Save results	Correlate by	Limit restarts	Restart	
		DNET581_DEMO_POLICY	✓			Host Name		✓	New F

Workflow Editor

Workflow components

Extensions

General activities

Wait until a situation is True

Take action or Write message

Change Policy Distribution

Assigned

DSNC:MVSA:DB2
WMQA:MVSA:MQESA
WMQB:MVSA:MQESA

Avail

.OMEG
BWF0
BWF1
CNM16
CXEGDSSR:MVSC:STORAGE
CXEGDSST:MVSA:STORAGE
CXEGN3:MVSA:KN3AGENT
CXEGN3:MVSB:KN3AGENT

Available Managed System Lists

*ALL_CMS
*CPIRA_MGR
*CUSTOM_UAGENT00
*CUSTOM_ZTWS00
*EIB
*GENERIC_CONFIG
*HUB
*LINUX_SYSTEM

Edit Managed System Lists

OK | Cancel | Help

action: LOG
EST ME...

Note – This scenario works because the managed systems involved are on the same host (MVSA)

Specify a managed system for the DB2 situation – in this example DSNC:MVSA:DB2

Specify managed systems for the MQ situation

Another Policy Example
Multiple Situations And Commands – Different Hosts

Workflows

Policy Details

Compatibility levels

Undo	Edit Workfl...	Policy name	Distributed	Auto start	Save results	Correlate by	Li	
		dnet581_demo_pol2	☑	☐	☐	Logical Application Group	☐	☑ New F

Correlate by Logical Application Group

Workflow Editor

Workflow components

Extensions | Name missing | General activities | Emitter activities

Wait until a situation is True

Evaluate a situation now

dnet581_demo_pol2 - Grapher View

Wait until Demo_DB2_Alert is True

Action succeeded

Wait until NT_Log_Space_Low is True

Check Windows alert

Check DB2 alert

Situation is true

Take action: LOG 'TEST ME...

Situation is true

Take action: LOG ' TEST M...

Issue command

Issue command

Distribution Of The Policy

Policy Details

Compatibility levels

Undo	Edit Workfl...	Policy name	Distributed	Auto start	Save results	Correlate by	Limit restarts	Restart
		dnet581_demo_pol2	✓			Logical Application Group		✓

Workflow Editor

Workflow components

Extensions | Name miss

General activities | Emitter ac

Wait until a situation is True

Evaluate a situation now

Change Policy Distribution

Assigned

TEST_LIST

Available Managed Systems

Available Managed System Lists

*ALL_CMS
*CPIRA_MGR
*CUSTOM_UAGENT00
*CUSTOM_ZTWS00
*EIB
*GENERIC_CONFIG
*HUB
*LINUX_SYSTEM

Edit Managed System Lists

OK | Cancel | Help

e action: LOG TEST M...

Apply | Help

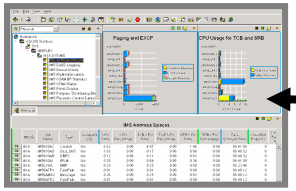
Edward A. Wood

Modify | Start/Stop

Distribution for this policy is to a Managed System List

Policies Require An Understanding Of The Monitoring Infrastructure

Distribution of policies requires a more complete understanding of the monitoring topology
- **What agents?**
- **Agents on what platforms?**
- **What TEMS?**
- **TEMS on what platforms?**



Portal Server TEPS

TEMS Remote

Windows
ITM TEMA

UNIX
ITM TEMA

Linux
ITM TEMA

OMEGAMON TEMA

OMEGAMON TEMA

TEMS HUB

z/OS

TEMS Remote

OMEGAMON TEMA

OMEGAMON TEMA

z/OS

A Common Scenario - Use Policies To Forward Alerts

Workflow Editor

Workflow components

Extensions | Name missing | General activities | Emitter activities

TEC_Event | ITO_Event | NetView_Event | SNMP Event

dnet581_SNMP - Grapher View

Wait until Demo_DB2_Alert is True → Situation is true → SNMP_Event4 SNMP Event

Emitter Settings

Emitter Type: SNMP Event

Parameters: Emitter target, Severity, Category, TrapDestination, Attributes

Current value: DB2 Thread Exceptions.Plan Name, DB2 Thread Exceptions.DB2ID

Operation Mode: Invoke emitter once for each data row / Bundle data rows and invoke emitter once

Policies may be used to forward alert information to TEC or to SNMP managers. This is one of the more common usages of policies.

Note – for z/OS hub TEMS a policy is needed to forward alerts to TEC. For non-z/OS hubs this limitation does not apply.

Pass information attributes as part of SNMP alert

Another Common Usage

Use Policies As An 'Overseer' For The Tivoli Monitoring Infrastructure

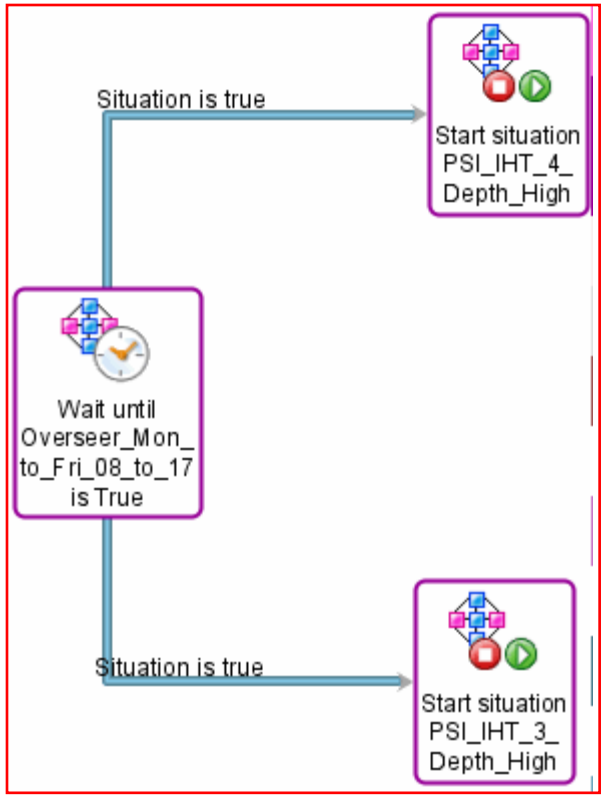
- Some situation alerts are sensitive to certain times of day or day of week considerations
  - This may be due to operational or off-hours processing concerns
  - Workloads will often vary during the day and during the week
  - Some issues are critical during prime time and not as critical off-hours
- 'Overseer' policies may be used to manage monitoring activity
  - Policies may be used to start/stop situations as needed based upon specified logic
    - Simplifies coding and maintenance in the underlying situations
  - Policies may be used to start/stop other policies as needed based upon specified logic
  - Optimize monitoring activity by running situations and policies on an 'as needed' basis

Example
Using Policies To Manage Time Of Day Sensitive Situations

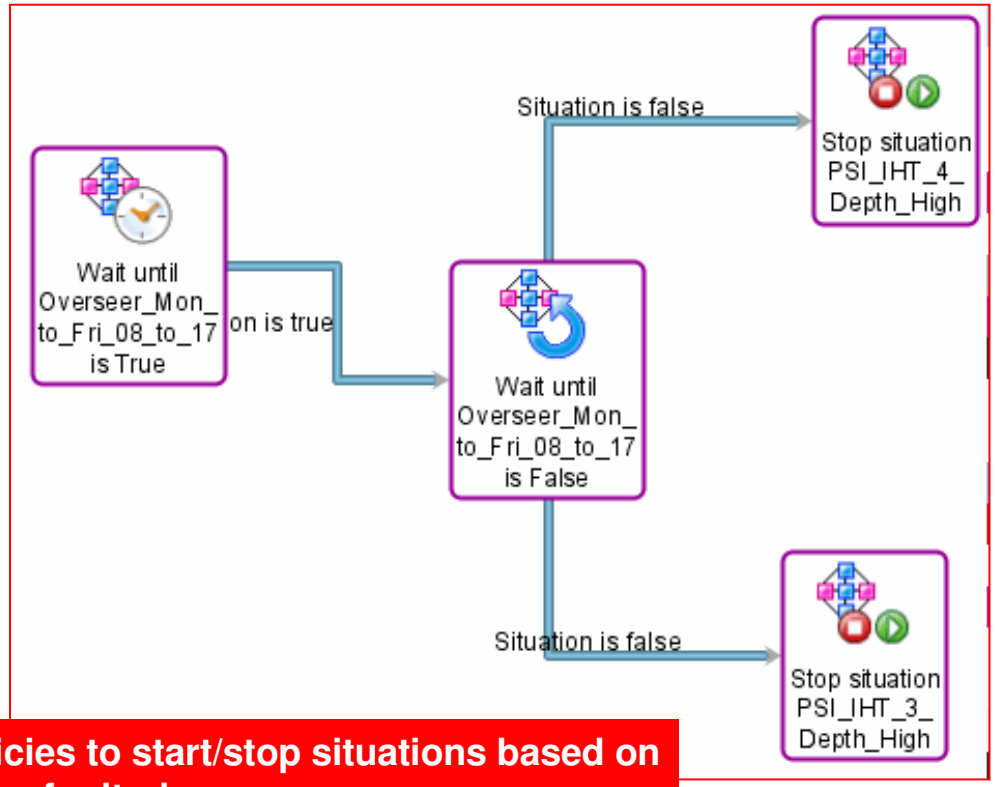
- Using policies to manage time of day sensitivity provides several advantages
  - Simplified situation formula (not cluttered with local_time predicates)
  - Increased space available for application attributes in the situation formula (Local_Time predicates can be complex and consume most of the formula)
  - Maintenance of the situations is greatly eased
  - Mixing application tables and LocalTime restricts certain features or situations
  - Reduced overhead: the situations are only running when needed

Using A Policy To Manage Situations Based Upon Time Of Day Requirements

Overseer policy to start situations



Overseer policy to stop situations



Use policies to start/stop situations based on a variety of criteria

- **Time of day**
- **Start specific analysis situations**
- **and more…..**

Policies
What They Are And What They Are Not

- Policies – What they are
  - Policies extend concepts established with situations and add additional functionality to the TEP
  - Policies expand the integrated command and control capabilities of the TEP
    - Situations remain the essential starting point
  - Policies add additional function and flexibility to situation capabilities
  - Policies may be used for basic alert correlation
- Policies – What they are not
  - The command capabilities of situations and policies are not a substitute for a full function automation engine
    - Policies and situations may be used to feed mechanisms such as SA/z
  - While policies may be used for basic alert correlation, policies are not a correlation engine
    - Understand the limitations of policies containing multiple situations
    - For high volume, complex event correlation consider other Tivoli solutions

Policies Versus Situations - Recommendations

- Use situations as the primary alert/command mechanism when possible
  - Situations will typically be more efficient than using policies
    - Well crafted situations using appropriate sampling intervals and boolean logic will be highly effective
  - Situations can usually run at the level of the agent
    - Policies must be executed within the TEMS infrastructure
    - Situations may exploit agent intelligent remote architecture
  - Situations will typically be easier to deploy
    - Policies will require more working knowledge of the monitoring topology and infrastructure
- Use policies to expand the capabilities of situations
  - When multiple commands required
  - When alerts to TEC or SNMP required
  - Use policies to make situation management more efficient

Policies – Recommendations And Best Practices

- Policies provide an exciting and powerful command and control facility integrated directly within Tivoli Enterprise Portal (TEP)
- Use policies where it is the most appropriate
  - Use policies when the job cannot be accomplished by situations alone
    - Example – the scenario requires multiple commands be issued
  - Use policies to optimize the monitoring and alerting
    - Example – use policies to activate situations when needed
  - Use a keep it simple methodology. Consider carefully before deploying large multi-situation policies across large numbers of managed systems
    - Test to make sure desired outcome is achieved
- Have a clear understanding of the agent/TEMS topology and requirements when deploying policies
  - Understand the subtleties of how policies operate

Policies – Recommendations And Best Practices

- There are many excellent uses for policies
  - Scenarios where a situation alert requires multiple commands be executed
    - Example – policy issues a corrective command and then issues a notification command to alert the corrective action was taken
  - Scenarios where monitoring information from multiple situations needs to be fed to automation
    - Example – multi-situation policy where commands are issued to pass alert information to console automation
  - Situations that need to pass alert information to other alert technologies
    - Example – alerts to TEC (depending upon TEMS topology), SNMP alerts
  - Use policies to optimize situation and monitoring usage
    - Example – policies to start certain higher cost situations only as needed
    - Example – policies to manage situations based upon time of day requirements
    - The benefit is to save the cost of ongoing alerting and monitoring

Summary - And Thank You

- Policies are an important component of the Tivoli Enterprise Portal and OMEGAMON DE
- Policies expand the integrated automation capabilities of the TEP
- When used effectively policies truly expand the power of the portal

- Thank you to my IBM colleagues for their very valuable input to this presentation
  - Richard Roy, Ken Stroble, Bay Van Horne, Joe Means, Barry Lamkin, Don Zeunert, Mike Stevens, Lih Wang

Thank You for Joining Us today!

Go to **www.ibm.com/software/systemz** to:

- Replay this teleconference
- Replay previously broadcast teleconferences
- Register for upcoming events