Chapter 1
Overview

What is the Upward Integration Option for Unicenter NSM?

The Upward Integration option for Unicenter NSM provides features that enable TNG to monitor machines with IBM Director Agents. The information can be collected across a widely dispersed enterprise network and made available in real-time to a single TNG console.

Upward Integration option for Unicenter NSM provides the following features:

- Auto-discovery of Director Agents
- "Director Agents" Business Process View
- Fault monitoring of individual Agents
- AMO

Introduction

The Upward Integration option for Unicenter integrates IBM Director Agent with Unicenter NSM. The agents are discovered and monitored by Unicenter DSM (Distributed State Machine) and WV (World View) components. All the trap messages sent by the agents are formatted and forwarded to the TNG console.

Upward Integration option for Unicenter NSM Components

Upward Integration option for Unicenter NSM consists of three main components:

Distributed State Machine (DSM) Component: The DSM component discovers the Director Agents. The DSM polls the agents and monitors their status.

World View (WV) component: The WV component creates a Business Process View for Director Agents. All the nodes monitored by the DSM in the network with the Director Agents are included in the Business Process view.

Asset Management Option Component: The Asset Management Option component integrates Director Agents with the AMO Agent of Unicenter NSM. If this option is installed, AMO agent detects Director Agents as an asset and will add their asset management data to its AMO agent database.

Integration with Unicenter NSM

The Upward Integration option for Unicenter adds new WorldView classes and instance level properties and automatically generates Business Process View.

Chapter 2
Architecture

The Upward Integration option for Unicenter consists of three main components: DSM Component, WV Component and AMO Component.

DSM Component:

This component consists of the following files:

- Policy files copied into AGENTWORKS_DIR\Services\Config\Aws_nsm\Dm directory
- Class files copied into AGENTWORKS_DIR\Services\Config\Aws_Wvgate directory
- Mib files copied into AGENTWORKS_DIR\Services\Config\Mibs directory
- Abrowser files copied into AGENTWORKS_DIR\Config\Abrowser

All the agents have their own policy, Class, Mib and Abrowser files and the files are loaded into Unicenter DSM as part of installation.

- The following Policy files are loaded:
  1. alertOnLan.atp
  2. bladesppalt.atp
  3. IbmAgent.atp
  4. ibmpsgLMSensor.atp
  5. ibmpsgmemoery.atp
  6. ibmpsgnic.atp
  7. ibmpsgpower.atp
  8. ibmpsgProcessor.atp
  9. ibmServeRaid.atp
  10. pET.atp
  11. umsEvent.atp
  12. win32WMI.atp
  13. wmalert.atp

Initially when awservices is started, Unicenter discovers all the hosts within the network. In the second level it discovers agents within the hosts. Once all the agents are discovered, the hosts with Director Agents are grouped and placed under a newly created Business Process View called "DIRECTOR_AGENTS". Each Director Agent host can be monitored by Node View.

The following World View Class files are loaded:

1. ibmpsgLMSensor.wvc
2. ibmpsgWin32WMI.wvc.
3. ibmpsgmemory.wvc
4. ibmpsgnic.wvc
5. ibmpsgpower.wvc
6. ibmpsgProcessor.wvc
7. ibmServeRaid.wvc

All of the above files create New Classes under the Agent Class.

World View Component:

This component consists of the following files:

- MIB files are copied into AGENTWORKS_DIR\Schema\Included
- Abrowser files are copied into AGENTWORKS_DIR\Config\Abrowser

The files are loaded into Unicenter Worldview as part of the installation.

Objects are created in 2D map by the Class files to represent the agents. The status of the objects in DSM and WV are synchronized.

AMO Agent Component:

This component consists of just one file, ums_inv2.bat. This file is copied into AMO Agent directory. Every asset detected by the AMO Agent has its own mif file; ums_inv2.bat creates the mif file for Director Agent's asset data. When AMO Agent runs, it calls umclient.bat. A line is added at the beginning of umclient.bat to call ums_inv2.bat.

Chapter 3
Upward Integration Option for Unicenter NSM - Method of Operation

The Upward Integration Option for Unicenter has been designed to take advantage of existing TNG infrastructure components such as the Distributed State Machine (DSM), the Common Object Repository and the WorldView map applications. Features provided by this option may be grouped into the following categories:

- Discovery of Director Agents and grouping them in a Business Process view
- Representation of Director Agents in the World View map applications
- Real time monitoring of IBM system components and event notification. NOTE: The System Health Monitoring install component in the Director Agent install must be selected as an option to get this function.

Discovery of Director Agents

Director Agent discovery is a two-step process as follows:

- All machines are discovered using the Auto Discovery capability of Unicenter TNG. These machines appear under the TCP/IP topology in the World View maps.

- DSM queries each of the Director Agent nodes for status and creates an object (in the object store) corresponding to each of the Director Agents. DSM also creates Director Agent object in the WV core for each of the Director Agents that is discovered in the network

Representation of Director Agents

- Director Agents are represented in Business Process View.
- Each machine with Director Agent is included in the Business Process View.

Real Time Monitoring and Event Notification

DSM periodically polls the Nodes with Director Agent to monitor the following entities:

- Temperature
- Voltage
- Tachometer
- Power Supply
- Physical Memory
- Processor
- Network Interface Card
- Physical Disks
- Logical Drives
- RAID subsystem

Events are reported to the EM console under the following conditions:

- If Director Agent fails to respond to DSM poll
- If the status of any of the above variables goes to Warning
- If the status of any of the above variables goes to Critical
- If the status of any of the above variable goes to Normal from Warning or Critical

Upward Integration Option for Unicenter Components

To address the issue of scalability in large distributed networks, the inherent design of Upward Integration Option for Unicenter consists of the following major software components:

1. **DSM policy to discover, monitor Director Agents**
2. **WorldView class definitions**
3. **Node View and Agent View applications**

Each of the above components is described below:

1. Upward Integration Option for Unicenter DSM policy

The Upward Integration Option for Unicenter DSM policy:

- Discovers Director Agents by polling the nodes
- Creates Business Process view and adds the nodes with Director Agents to Business Process view
- Changes the state of Director Agents objects in Common Object Repository.
- Listens to traps from Director Agents and updates the status of the Agent object
- Periodically polls the Director Agent nodes to monitor the status of the Agents

2. Upward Integration Option for Unicenter WorldView Class Definitions

Upward Integration Option for Unicenter creates new WorldView classes.

ibmpsgLMSensor
ibmpsgWin32WMI
ibmpsgProcessor
ibmpsgMemory
ibmpsgNIC
ibmpsgPower
ibmServeRaid

Note: The above classes will be loaded into the TNG Repository during installation and class definition may be viewed using the class wizard.

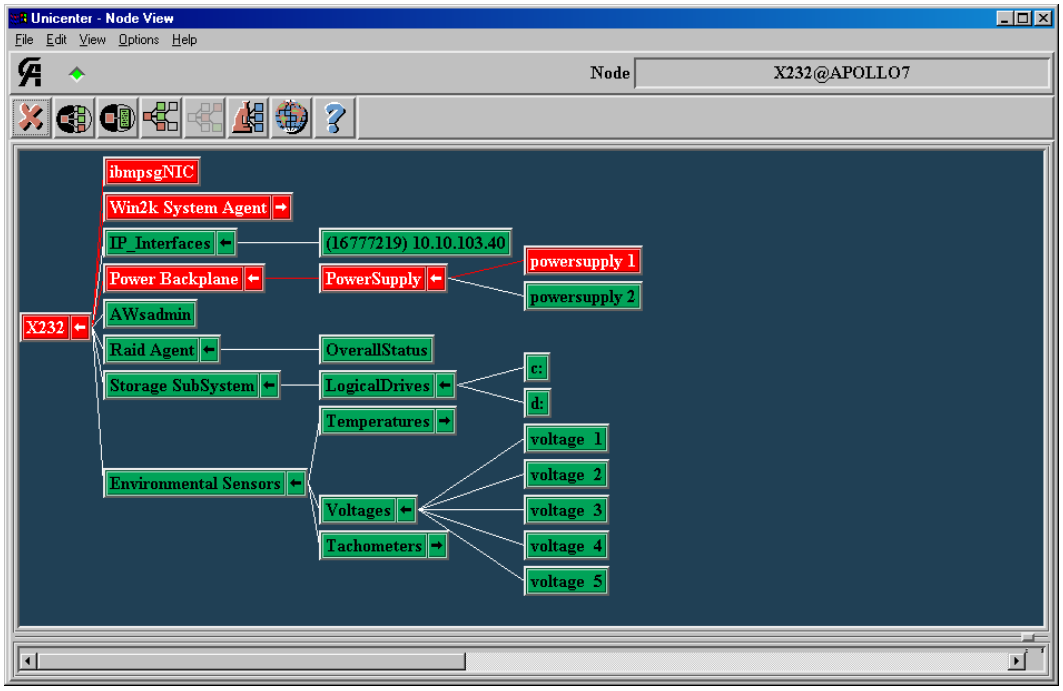
3. Node View and Agent View applications

Node View Application

The standard Node View application can be launched in context using one or more methods.

- Launched from a Director Agent object in the Director Agent BPV
- Launched from DSM View

The Node View application for a Director Agent is as shown below:



The Node View display for Environmental Sensor Agent may contain one or more dynamic objects. These are displayed as children of Environmental Sensor Agent object in Node View and represent one or more of the following monitored entities:

- State of Temperature Sensor as represented by the MIB variable "ibmpsgTemperatureSensorStatus". Values for the "ibmpsgTemperatureSensorStatus" are Normal, Warning and Critical.
- State of VoltageSensor as represented by the MIB variable "ibmpsgVoltageSensorStatus". Values for the "ibmpsgVoltageSensorStatus" are Normal, Warning and Critical.

- State of TachometerSensor as represented by the MIB variable "ibmpsgTachometerStatus". Values for the "ibmpsgTachometerStatus" are Normal, Warning and Critical.

The Node View display for a StorageSubSystem Agent may contain one or more dynamic objects. These are displayed as children of StorageSubSystem Agent object in Node View and represent one or more of the following monitored entities:

- State of the Logical Drives as represented by the MIB variable "Win32LogicalDiskStatus". Values for the "Win32LogicalDiskStatus" are Normal, Warning and Critical.
- State of the Physical Drives as represented by the MIB variable "Win32DiskDriveStatus". Values for the "Win32DiskDriveStatus" are Normal, Warning and Critical.

The Node View display for a Power Supply Agent may contain one or more dynamic objects. These are displayed as children of Power Supply Agent object in Node View and represent one or more of the following monitored entities:

- State of the Power Supply as represented by the MIB variable "ibmpsgPowerSupplyStatus". Values for the " ibmpsgPowerSupplyStatus" are Normal, Warning and Critical.

The Node View display for a Physical Memory Agent may contain one or more dynamic objects. These are displayed as children of Physical Memory Agent object in Node View and represent one or more of the following monitored entities:

- State of the Physical Memory as represented by the MIB variable "ibmpsgPhysicalMemoryStatus". Values for the "ibmpsgPhysicalMemoryStatus" are Normal, Warning and Critical.

The Node View display for a Network Subsystem Agent may contain one or more dynamic objects. These are displayed as children of

Network Subsystem Agent object in Node View and represent one or more of the following monitored entities:

- State of the Network Subsystem as represented by the MIB variable "ibmpsgPhysicalNetworkAdapterStatus". Values for the " ibmpsgPhysicalNetworkAdapterStatus" are Normal, Warning and Critical.

The Node View display for a Processor Agent may contain one or more dynamic objects. These are displayed as children of Processor Agent object in Node View and represent one or more of the following monitored entities:

- State of the Processor as represented by the MIB variable "ibmpsgProcessorStatus". Values for the " ibmpsgProcessorStatus" are Normal, Warning and Critical.

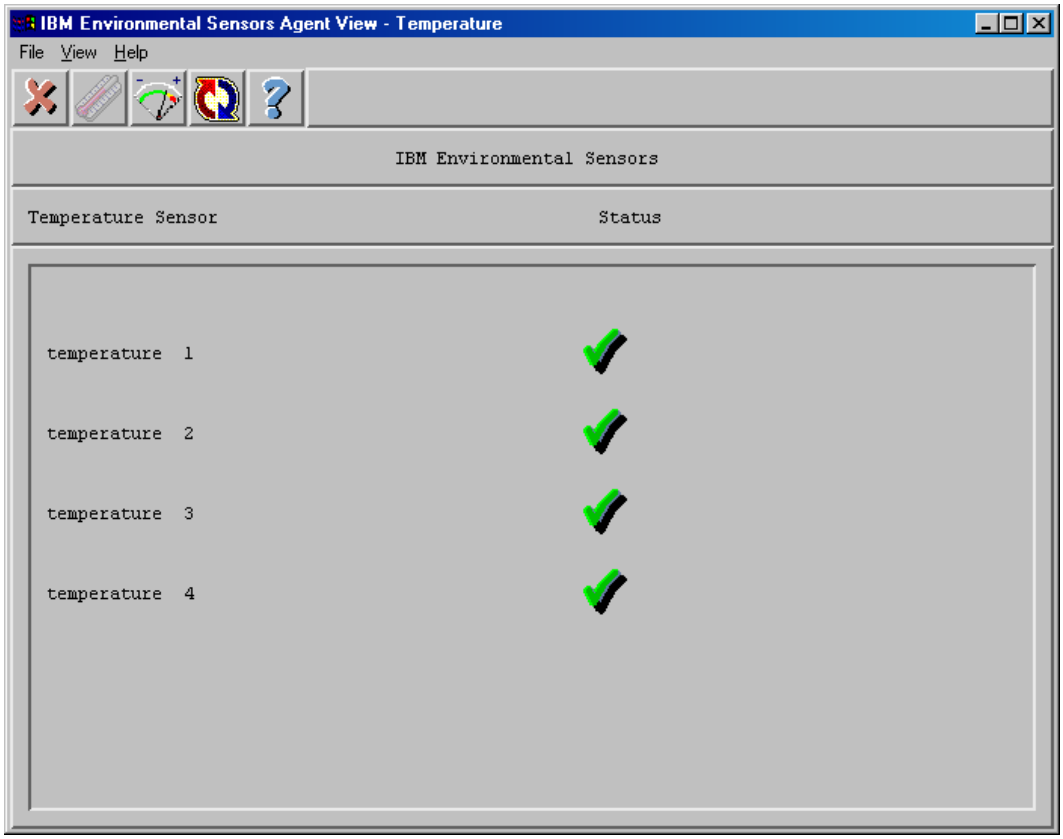
The Node View display for a Raid Agent may contain one or more dynamic objects. These are displayed as children of Raid Agent object in Node View and represent one or more of the following monitored entities:

- State of the Raid as represented by the MIB variable "ibmpsgUMSComponenetHealthCurrentState". Values for the " ibmpsgUMSComponenetHealthCurrentState" are Normal, Warning and Critical.

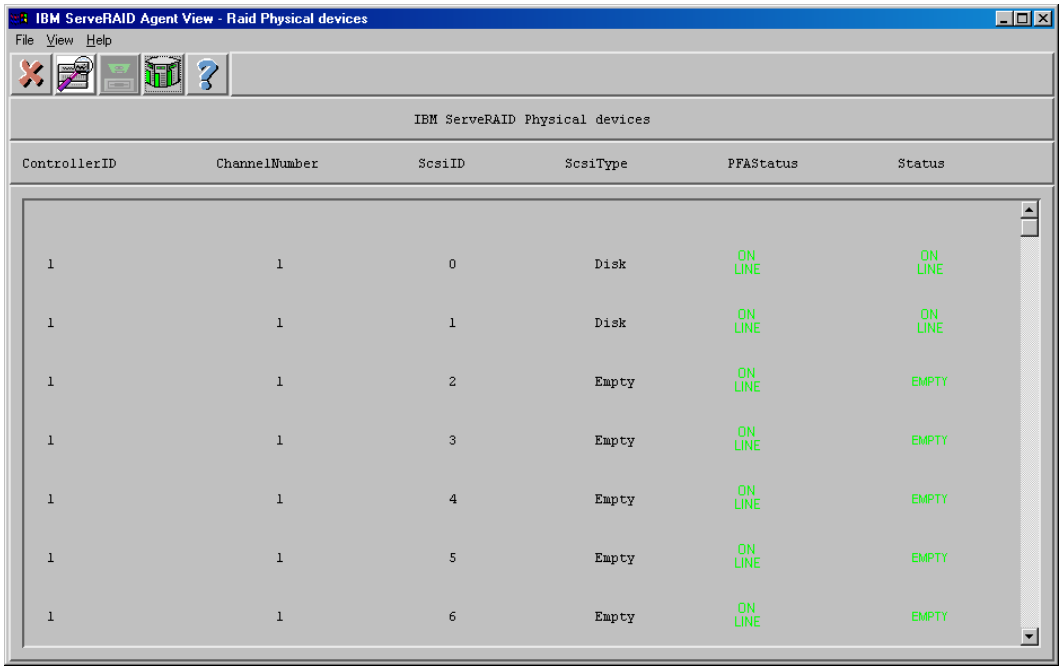
Agent View Applications

Two new Agent View applications are included with the Upward Integration Option for Unicenter. These applications are launched in context from Node View Menu items of Environmental Sensor and Raid objects.

Shown below is an Agent View of Environmental Sensor



Shown below is an Agent View of Raid



Note:
The following policies listen to traps only and on receiving traps, they are converted to Events and forwarded to EM Console.

- umsevent
- aolntrap
- aolnpet
- wmalert

By default EM listens to traps. If EM and DSM are on the same box, then the console will have two events reported for every trap received, to overcome the

problem EM can be asked to drop the traps (with specific oids) that the polices are listening to by updating the file TNG/caiuser/Catrapd.cfg.

Appendix

Policy Details

We have the policy files for the following Agents

1. Raid
2. Environmental Sensors
3. Storage Sub System
4. Memory
5. Network Sub System
6. Processor
7. Power Supply
8. Umsevent
9. aolntrap
10. aolnpet
11. wmalert
12. bladesppalt

Raid:

This MIB represents a real agent and is represented in WV. The policy (ibmServeRaid.atp) creates a new class (ibmServeRaid) under the Agent class. The policy polls the Agent for MIB variables and listens to the traps as well. DSM sends state change events to the Event Management Console and trap messages are processed and converted to Events and forwarded to EM Console. The Director Agents web-based interface can be launched from WordView menu. ibmServeRaid MIB is loaded in TNG with MIB name ibmServeRaid.

The policy listens to the following traps:

Enterprise OID	Generic trap	Specific trap	Description
1.3.6.1.4.1.2.6.167.2	6	201	Informational: No controllers were found in this system
1.3.6.1.4.1.2.6.167.2	6	202	Error: Commands not responding on Controller
1.3.6.1.4.1.2.6.167.2	6	203	Error: The battery-backup cache device on Controller needs a new battery
1.3.6.1.4.1.2.6.167.2	6	204	Error: The battery-backup cache device on Controller is defective
1.3.6.1.4.1.2.6.167.2	6	205	Error: Background polling commands not responding on Controller
1.3.6.1.4.1.2.6.167.2	6	206	Error: Cannot read controller configuration
1.3.6.1.4.1.2.6.167.2	6	207	Informational: Controller has been added to the system
1.3.6.1.4.1.2.6.167.2	6	208	Informational: Controller has been replaced in the system
1.3.6.1.4.1.2.6.167.2	6	209	Informational: Controller failover detected. Passive controller is now active
1.3.6.1.4.1.2.6.167.2	6	301	Warning: Logical Drive is Critical on Controller
1.3.6.1.4.1.2.6.167.2	6	302	Error: Logical Drive is Blocked on Controller
1.3.6.1.4.1.2.6.167.2	6	303	Error: Logical Drive is Offline on Controller
1.3.6.1.4.1.2.6.167.2	6	304	Informational: Rebuilding Logical Drive on Controller
1.3.6.1.4.1.2.6.167.2	6	305	Informational: Rebuild complete on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	306	Error: Rebuild failed on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	307	Informational: Synchronizing

			Logical Drive on Controller
1.3.6.1.4.1.2.6.167.2	6	308	Informational: Synchronization complete on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	309	Error: Synchronization failed on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	310	Informational: Migrating Logical Drive on Controller
1.3.6.1.4.1.2.6.167.2	6	311	Informational: Migration complete on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	312	Error: Migration failed on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	313	Informational: Compressing Logical Drive on Controller
1.3.6.1.4.1.2.6.167.2	6	314	Informational: Compression complete on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	315	Error: Compression failed on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	316	Informational: Decompressing Logical Drive on Controller
1.3.6.1.4.1.2.6.167.2	6	317	Informational: Decompression complete on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	318	Error: Decompression failed on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	319	Informational: FlashCopying Logical Drive on Controller
1.3.6.1.4.1.2.6.167.2	6	320	Informational: FlashCopy complete on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	321	Error: FlashCopy failed on Logical Drive of Controller
1.3.6.1.4.1.2.6.167.2	6	322	Informational: Rebuilding Array

			on Controller
1.3.6.1.4.1.2.6.167.2	6	323	Informational: Rebuild complete on Array of Controller
1.3.6.1.4.1.2.6.167.2	6	324	Error: Rebuild failed on Array of Controller
1.3.6.1.4.1.2.6.167.2	6	325	Informational: Synchronizing Array on Controller
1.3.6.1.4.1.2.6.167.2	6	326	Informational: Synchronization complete on Array of Controller
1.3.6.1.4.1.2.6.167.2	6	327	Error: Synchronization failed on Array of Controller
1.3.6.1.4.1.2.6.167.2	6	328	Informational: FlashCopying Array on Controller
1.3.6.1.4.1.2.6.167.2	6	329	Informational: FlashCopy complete on Array of Controller
1.3.6.1.4.1.2.6.167.2	6	330	Error: FlashCopy failed on Array of Controller
1.3.6.1.4.1.2.6.167.2	6	401	Error: A drive is Defunct on Controller
1.3.6.1.4.1.2.6.167.2	6	402	Warning: PFA detected on Controller
1.3.6.1.4.1.2.6.167.2	6	403	Informational: A drive is set to Hot-Spare on Controller
1.3.6.1.4.1.2.6.167.2	6	501	Informational: Enclosure device responding on Controller
1.3.6.1.4.1.2.6.167.2	6	502	Error: Enclosure device not responding on Controller
1.3.6.1.4.1.2.6.167.2	6	503	Informational: Enclosure fan on Controller, Channel is now operational
1.3.6.1.4.1.2.6.167.2	6	504	Error: Enclosure fan on Controller, Channel is malfunctioning
1.3.6.1.4.1.2.6.167.2	6	505	Informational: Enclosure fan on

			Controller, Channel has been installed
1.3.6.1.4.1.2.6.167.2	6	506	Warning: Enclosure fan on Controller, Channel has been removed
1.3.6.1.4.1.2.6.167.2	6	507	Error: Enclosure temperature is in normal range on Controller
1.3.6.1.4.1.2.6.167.2	6	508	Error: Enclosure temperature is out of normal range on Controller
1.3.6.1.4.1.2.6.167.2	6	509	Informational: Enclosure power supply on Controller is now operational
1.3.6.1.4.1.2.6.167.2	6	510	Error: Enclosure power supply on Controller, Channel is malfunctioning
1.3.6.1.4.1.2.6.167.2	6	511	Informational: Enclosure power supply on Controller, Channel has been installed
1.3.6.1.4.1.2.6.167.2	6	512	Warning: Enclosure power supply on Controller, Channel has been removed

Environmental Sensors:

This MIB represents a real agent and is represented in WV. The policy (ibmpsgLMSensor.atp) creates a new class (ibmpsgLMSensor) under the Agent class. The policy polls the Agent for MIB variables and listens to the traps as well. DSM sends state change events to the Event Management Console and trap messages are processed and converted to Events and forwarded to EM Console. The Director Agents web-based interface can be launched from WorldView menu, umslmsensor MIB is loaded in TNG with MIB name ibmpsgLMSensor.

The MIB has the following tables and each table is represented as class.

ibmpsgTempSensors *:*

ibmpsgTempSensors (represents the table iBMPSGTemperatureSensorTable),
ibmpsgTempSensor (represents the rows within the table)
ibmpsgTempSensors listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	2

(Above trap is Temperature Event defined in umsevent.mib)
ibmpsgTempSensor polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID)

***ibmpsgVolSensors**:*
ibmpsgVolSensors (represents the table iBMPSGVoltageSensorTable)
ibmpsgVolSensor (represents the rows with in the table)
ibmpsgVolSensors listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	3

(Above trap is Voltage Event defined in umsevent.mib)
ibmpsgVolSensor polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID)

***ibmpsgTachometers**:*
ibmpsgTachometers (represents the table iBMPSGTachometerTable)
ibmpsgTachometer (represents the rows with in the table).
ibmpsgTachometers listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	5

ibmpsgTachometer polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID)

Abrowser Script for Environmental Sensors:

Temperature status:

Go Critical if ibmpsgTemperatureSensorCurrentReading exceeds ibmpsgTemperatureSensorLowerThresholdCritical or go Warning if ibmpsgTemperatureSensorCurrentReading exceeds ibmpsgTemperatureSensorLowerThresholdNonCritical.

Voltage Status:
Go Critical only if ibmpsgVoltageSensorCurrentReading is less than ibmpsgVoltageSensorLowerThresholdNonCritical or greater than ibmpsgVoltageSensorUpperThresholdNonCritical.

Tachometer Status*:*
Go Critical if ibmpsgTachometerCurrentReading is less than ibmpsgTachometerLowerThresholdCritical or go Warning if ibmpsgTachometerCurrentReading is less than ibmpsgTachometerLowerThresholdNonCritical.

StorageSubsystem:
This MIB represents a real agent and is represented in WV. The policy (win32WMI.atp) creates a new class (ibmpsgWin32WMI) under the Agent class. The policy polls the Agent for MIB variables and listens to the traps as well. DSM sends state change events to the Event Management Console and trap messages are processed and converted to Events and forwarded to EM Console. The Director Agents web-based interface can be launched from WordView menu. StorageSubsystem MIB is loaded in TNG with MIB name win32WMI.

The MIB has the following tables and each table is represented as class:

ibmpsgDiskDrives*:*
ibmpsgDiskDrives (represents the table win32DiskDriveTable),
ibmpsgDiskDrive (represents the rows within the table)
ibmpsgDiskDrives does not poll and listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	9

(Above trap is Smart Event defined in umsevent.mib)

ibmpsgDiskDrives polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID).

***ibmpsgLogicalDisks**:*
ibmpsgLogicalDisks (represents the table win32LogicalDiskTable),
ibmpsgLogicalDisk (represents the rows with in the table)
ibmpsgLogicalDisks listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	7

(Above trap is Storage Event defined in umsevent.mib)

ibmpsgLogicalDisk polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID)

Physical Memory:
This MIB represents a real agent and is represented in WV. The policy (ibmpsgmemory.atp) creates a new class (ibmpsgmemory) under the Agent class. The policy polls the Agent for MIB variables and listens to the traps as well. DSM sends state change events to the Event Management Console and trap messages are processed and converted to Events and forwarded to EM Console. The Director Agents web-based interface can be launched from WorldView menu. Physical Memory MIB is loaded in TNG with MIB name ibmpsgMemory.

The MIB has the following tables and each table is represented as class

***ibmpsgMemorySensors**:*
ibmpsgMemorySensors (represents the table ibmpsgPhysicalMemoryTable),
ibmpsgMemorySensor (represents the rows with in the table)
ibmpsgMemorySensor listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	19

(Above trap is Memory Event defined in umsevent.mib)
ibmpsgMemorySensor polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID)

Network Subsystem:
This MIB represents a real agent and is represented in WV. The policy (ibmpsgnic.atp) creates a new class (ibmpsgNIC) under the Agent class. The policy polls the Agent for MIB variables and listens to the traps as well. DSM sends state change events to the Event Management Console and trap messages are processed and converted to Events and forwarded to EM Console. The Director Agents web-based interface can be launched from WordView menu. Network Subsystem MIB is loaded in TNG with MIB name ibmpsgNIC.

The MIB has the following tables and each table is represented as class.

ibmpsgPhyNicSensors*:*
ibmpsgPhyNicSensors (represents the table ibmpsgPhysicalNetworkAdapterTable),
ibmpsgPhyNicSensor (represents the rows within the table)
ibmpsgPhyNicSensor listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	26
1.3.6.1.4.1.2.6.159.1.1	6	27
1.3.6.1.4.1.2.6.159.1.1	6	28

(Above trap is Network adapter Event defined in umsevent.mib)
ibmpsgPhyNicSensor polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID)

Processor:
This MIB represents a real agent and is represented in WV. The policy (ibmpsgProcessor.atp) creates a new class (ibmpsgProcessor) under the Agent class. The policy polls the Agent for MIB variables and listens to the traps as

well. DSM sends state change events to the Event Management Console and trap messages are processed and converted to Events and forwarded to EM Console. The Director Agents web-based interface can be launched from WordView menu. Processor MIB is loaded in TNG with MIB name ibmpsgProcessor.

The MIB has the following tables and each table is represented as class.

***ibmpsgProcessorSensors**:*
ibmpsgProcessorSensors (represents the table ibmpsgProcessorTable),
ibmpsgProcessorSensor (represents the rows with in the table)
ibmpsgProcessorSensors listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	18

(Above trap is Processor Event defined in umsevent.mib)
ibmpsgProcessorSensor polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID)

Power Supply:
This MIB represents a real agent and is represented in WV. The policy (ibmpsgpower.atp) creates a new class (ibmpsgPower) under the Agent class. The policy polls the Agent for MIB variables and listens to the traps as well. DSM sends state change events to the Event Management Console and trap messages are processed and converted to Events and forwarded to EM Console. The Director Agents web-based interface can be launched from WorldView menu. Power Supply MIB is loaded in TNG with MIB name ibmpsgPower.

The MIB has the following tables and each table is represented as class.

***ibmpsgPowerSensors**:*
ibmpsgPowerSensors (represents the table ibmpsgPowerSupplyTable),
ibmpsgPowerSensor (represents the rows with in the table)
ibmpsgPowerSensors listens to the following trap:

Enterprise OID	Generic-trap	specific trap
1.3.6.1.4.1.2.6.159.1.1	6	23

(Above trap is Power Supply Event defined in umsevent.mib)
ibmpsgPowerSensor polls the Agent for row variables (converts the row index to oid format and appends the same to "1.3.6.1.4.1.2.6.159.1.1.30.3.1.2.", polls the agent for the new OID)

umsevent:
This MIB consists of trap definitions only and does not represent a real agent hence no representation in Worldview. The policy (umsEvent.atp) listens to traps and on receiving the traps messages are converted to Events and forwarded to EM Console. The traps are defined in umsevent MIB that is loaded in TNG with mibname ibmpsgEvent.

The policy listens to the following traps:

Enterprise OID	Generic trap	Specific trap	Description
1.3.6.1.4.1.2.6.159.1.1	6	12	ibmpsgLANLeashEventBindings
1.3.6.1.4.1.2.6.159.1.1	6	15	ibmpsgRedundantNetworkAdapter EventBindings
1.3.6.1.4.1.2.6.159.1.1	6	16	ibmpsgRedundantNetworkAdapter SwitchoverEventBindings
1.3.6.1.4.1.2.6.159.1.1	6	17	ibmpsgRedundantNetworkAdapter SwitchbackEventBindings

aolntrap:
This MIB consists of trap definitions only and does not represent a real agent hence no representation in Worldview.
The policy (alertOnLan.atp) listens to the traps and on receiving these traps the messages are converted to Events and forwarded to EM Console. The traps are defined in aolntrap MIB that is loaded in TNG with mibname alert_on_LAN.

The policy listens to the following traps:

Enterprise OID	Generic trap	Specific trap	Description
1.3.6.1.4.1.343.2.8.1.1	6	0	Presence Heartbeat Expired
1.3.6.1.4.1.343.2.8.1.1	6	1	Cover Tamper
1.3.6.1.4.1.343.2.8.1.1	6	2	Voltage/Fan/Temperature Out of

			Specification
1.3.6.1.4.1.343.2.8.1.1	6	3	LAN Leash Tamper
1.3.6.1.4.1.343.2.8.1.1	6	4	Temperature Out of Specification
1.3.6.1.4.1.343.2.8.1.1	6	5	Processor Missing
1.3.6.1.4.1.343.2.8.1.1	6	6	Processor Over Temperature
1.3.6.1.4.1.343.2.8.1.1	6	7	Watchdog Event
1.3.6.1.4.1.343.2.8.1.1	6	8	P.O.S.T.
1.3.6.1.4.1.343.2.8.1.1	6	9	Unknown Event
1.3.6.1.4.1.343.2.8.1.1	6	10	Processor 0 Missing
1.3.6.1.4.1.343.2.8.1.1	6	11	Processor 1 Missing
1.3.6.1.4.1.343.2.8.1.1	6	12	Voltage/Fan Out of Specification
1.3.6.1.4.1.343.2.8.1.1	6	13	Voltage Out of Specification
1.3.6.1.4.1.343.2.8.1.1	6	14	Fan Out of Specification
1.3.6.1.4.1.343.2.8.1.1	6	15	Fan/Temperature Out of Specification
1.3.6.1.4.1.343.2.8.1.1	6	16	Voltage/Temperature Out of Specification
1.3.6.1.4.1.343.2.8.1.1	6	17	Surprised Undock event
1.3.6.1.4.1.343.2.8.1.1	6	18	Event clear
1.3.6.1.4.1.343.2.8.1.1	6	19	Client added
1.3.6.1.4.1.343.2.8.1.1	6	20	Client deleted

aolnpet:
This MIB consists of trap definitions only and does not represent a real agent, hence no representation in Worldview.
The policy (pET.atp) listens to the traps and on receiving these traps the messages are converted to Events and forwarded to EM Console. The traps are defined in aolnpet MIB that is loaded in TNG with mibname pET_version_1.

The policy listens to the following traps:

Enterprise OID	Generic trap	Specific trap	Description
1.3.6.1.4.1.3183.1.1	6	2584320	Presence Heartbeat Expired
1.3.6.1.4.1.3183.1.1	6	356096	Cover Tamper
1.3.6.1.4.1.3183.1.1	6	552706	Voltage/Fan/Temperature Out of Specification
1.3.6.1.4.1.3183.1.1	6	356100	LAN Leash Tamper

1.3.6.1.4.1.3183.1.1	6	93952	Temperature Out of Specification
1.3.6.1.4.1.3183.1.1	6	356099	Processor Missing
1.3.6.1.4.1.3183.1.1	6	487169	Processor Over Temperature
1.3.6.1.4.1.3183.1.1	6	1142534	Watchdog Event
1.3.6.1.4.1.3183.1.1	6	1011456	P.O.S.T.
1.3.6.1.4.1.3183.1.1	6	159488	Voltage Out of Specification
1.3.6.1.4.1.3183.1.1	6	290560	Fan Out of Specification
1.3.6.1.4.1.3183.1.1	6	683778	Fan/Temperature Out of Specification
1.3.6.1.4.1.3183.1.1	6	356101	Surprised Undock event
1.3.6.1.4.1.3183.1.1	6	1076994	Event clear
1.3.6.1.4.1.3183.1.1	6	2277391	Alert on LAN 2 Event has occurred

wmalert:
This MIB consists of trap definitions only and does not represent a real agent hence no representation in Worldview.
The policy (wmalert.atp) listens to the traps and on receiving these traps the messages are converted to Events and forwarded to EM Console. The traps are defined in wmalert MIB that is loaded in TNG with MIB name nfsptrapg.

The policy listens to the following traps:

Enterprise OID	Generic trap	Specific trap	Description
1.3.6.1.4.1.2.6.158.1.1	6	0	Critical Alert: Temperature threshold exceeded
1.3.6.1.4.1.2.6.158.1.1	6	1	Critical Alert: Voltage threshold exceeded
1.3.6.1.4.1.2.6.158.1.1	6	2	Critical Alert: Remote login attempts threshold exceeded
1.3.6.1.4.1.2.6.158.1.1	6	3	Critical Alert: Multiple fan failure
1.3.6.1.4.1.2.6.158.1.1	6	4	Critical Alert: Power supply failure
1.3.6.1.4.1.2.6.158.1.1	6	5	Critical Alert: Hard disk drive failure
1.3.6.1.4.1.2.6.158.1.1	6	6	Critical Alert: Voltage Regulator Module (VRM) failure
1.3.6.1.4.1.2.6.158.1.1	6	10	Non-Critical Alert: Redundant Power Supply failure
1.3.6.1.4.1.2.6.158.1.1	6	11	Non-Critical Alert: Single Fan

			failure
1.3.6.1.4.1.2.6.158.1.1	6	12	Non-critical Error: Temperature threshold exceeded
1.3.6.1.4.1.2.6.158.1.1	6	13	Non-Critical Alert: Voltage threshold exceeded
1.3.6.1.4.1.2.6.158.1.1	6	15	System Alert: Secondary Device warning
1.3.6.1.4.1.2.6.158.1.1	6	20	System Alert: Post Timeout value exceeded
1.3.6.1.4.1.2.6.158.1.1	6	21	System Alert: Post Timeout value exceeded
1.3.6.1.4.1.2.6.158.1.1	6	22	System Alert: Application Alert
1.3.6.1.4.1.2.6.158.1.1	6	23	System Alert: Power Off
1.3.6.1.4.1.2.6.158.1.1	6	24	System Alert: Power On
1.3.6.1.4.1.2.6.158.1.1	6	25	System Alert: System Boot Failure
1.3.6.1.4.1.2.6.158.1.1	6	26	System Alert: OS Loader Timeout
1.3.6.1.4.1.2.6.158.1.1	6	27	System Alert: Predictive Failure Analysis(PFA) information

bladesppalt:
This MIB consists of trap definitions only and does not represent a real agent hence no representation in Worldview.
The policy (bladesppalt.atp) listens to the traps and on receiving these traps the messages are converted to Events and forwarded to EM Console. The traps are defined in bladesppalt MIB that is loaded in TNG with MIB name bladesppalt.

The policy listens to the following traps:

Enterprise OID	Generic trap	Specific trap	Description
1.3.6.1.4.1.2.6.158	6	0	Critical Alert: Temperature threshold exceeded
1.3.6.1.4.1.2.6.158	6	1	Critical Alert: Voltage threshold exceeded
1.3.6.1.4.1.2.6.158	6	2	Critical Alert: Remote login attempts

			threshold exceeded
1.3.6.1.4.1.2.6.158	6	3	Critical Alert: Multiple fan failure
1.3.6.1.4.1.2.6.158	6	4	Critical Alert: Power supply failure
1.3.6.1.4.1.2.6.158	6	5	Critical Alert: Hard disk drive failure
1.3.6.1.4.1.2.6.158	6	6	Critical Alert: Voltage Regulator Module (VRM) failure
1.3.6.1.4.1.2.6.158	6	10	Non-Critical Alert: Redundant Power Supply failure
1.3.6.1.4.1.2.6.158	6	11	Non-Critical Alert: Single Fan failure
1.3.6.1.4.1.2.6.158	6	12	Non-critical Error: Temperature threshold exceeded
1.3.6.1.4.1.2.6.158	6	13	Non-Critical Alert: Voltage threshold exceeded
1.3.6.1.4.1.2.6.158	6	15	System Alert: Secondary Device warning
1.3.6.1.4.1.2.6.158	6	20	System Alert: Post Timeout value exceeded
1.3.6.1.4.1.2.6.158	6	21	System Alert: Post Timeout value exceeded
1.3.6.1.4.1.2.6.158	6	22	System Alert: Application Alert
1.3.6.1.4.1.2.6.158	6	23	System Alert: Power Off
1.3.6.1.4.1.2.6.158	6	24	System Alert: Power On
1.3.6.1.4.1.2.6.158	6	25	System Alert: System Boot Failure
1.3.6.1.4.1.2.6.158	6	26	System Alert: OS Loader Timeout
1.3.6.1.4.1.2.6.158	6	27	System Alert: Predictive Failure Analysis(PFA) information
1.3.6.1.4.1.2.6.158	6	31	Critical Alert: Multiple switch module failure.
1.3.6.1.4.1.2.6.158	6	32	Non-Critical Alert: Redundant module.
1.3.6.1.4.1.2.6.158	6	33	System Alert: Keyboard/Video/Mouse(KVM) or Medial Tray(MT) switching failure.

1.3.6.1.4.1.2.6.158	6	34	System Alert: Inventory
1.3.6.1.4.1.2.6.158	6	35	System Alert: System Log 75% full.
1.3.6.1.4.1.2.6.158	6	36	Critical Alert: Incompatible hardware configuration.
1.3.6.1.4.1.2.6.158	6	37	System Alert: Network change notification.

EM Console Message format:
The format of state change message in EM Console is different if the policy is in ATP format.
A sample message if policy in ATP:
Host: Windows2000_Server Windows2000_Server ibmpsgLMSensor Policy ibmpsgTempSensor Critical Warning
deviceid="temperature 0"

A sample message if policy in CNF and DAT:
Host: Windows2000_Server Windows2000_Server ibmpsgLMSensor Poll ibmpsgTempSensor Critical Warning
deviceid="temperature 0"

In CNF and DAT format we can distinguish if the status change is due to Poll or Trap where as in ATP it just prints Policy in place of Trap or Poll.

The bladesppalt policy is an exception to this. It will use one of the following styles dependant on if the trap is from a BladeCenter Management Module or a standard server Remote Supervisor Adapter:

Host:Management_Module IBM_BladeCenter bladesppalt Trap Agent:bladesppalt Unknown Critical Physical intrusion of system trap received with message *

Host:Remote_Supervisor_Adapter IBM_Server Trap Agent:bladesppalt Unknown Critical Physical intrusion of system trap received with message *

Director Agent's Web-based interface:
The Director Agent's Web-based interface can be launched from WordView menu by right clicking on the object and selecting Director Agent. By default the Upward Integration Web-based interface is assigned port 411 that is the default port for Director Agent's Web-based interface. If the user opts for a different port for Director Agent's Web-based interface during Director Agent installation, then the user has to assign the same port to Upward Integration Web-based interface by manually changing the port number in the file AGENTWORKS_DIR\bin\umservices.bat.

Steps to create banner for Trap Messages received in Enterprise Management Console:

A message is defined by its message ID; message ID should be set the same as message you receive in Enterprise Management Console(EM).

Example: Generate a banner for a trap sent by the Agent machine to notify the status of temperature 1 as critical. The trap sends the following message to EM:

Host: Windows2000_Server Windows2000_Server ibmpsgLMSensor Policy ibmpsgTempSensor Up Critical temperature 1

1. **Define the message in Message Record**.
Open Messages from
Start->Program->Unicenter TND->EnterpriseManagement->EM Classic->WindowsNT->Event.
Click on Message then New in the menu. The Message Record – Details window comes up.
In the message ID Box specify entire text of message received, i.e.
Host: Windows2000_Server Windows2000_Server ibmpsgLMSensor Policy ibmpsgTempSensor Up Critical temperature 1

The Description Box is for detailed description of message. That is optional.
Click on File then Save.

Click on Actions, click Selected then New.
The Message Record Action - Detail window comes up.

2. **Define the corresponding action**.

More than one action can be defined for a particular message, the sequence number specifies the order for these actions. Set the sequence number as '*' so that the action has highest priority. Select the action as BANNER and specify the text to be displayed in the Text Box. There are a number of other options/specifications available in Message Action that you can set.

3. Load the Message Record:

Go to Enterprise Console and use 'opreload' command to load the new Message Record in the database. This command is to be used from Command Box in EM Console.

Send the trap. The formatted trap message is received in EM Console and also the corresponding banner defined is displayed.