Integrating ITM 6.1 with NetView 7.1.5

Version 2.0

May 2007

Jane Curry

Skills 1st Ltd

www.skills-1st.co.uk

Synopsis

This paper describes the new integration between IBM Tivoli Monitoring 6.1 and IBM Tivoli NetView 7.1.5. Integration with the IBM Tivoli Enterprise Console 3.9 is also discussed.

Initially, a very brief overview of NetView and ITM 6.1 are presented for those who are unfamiliar with either product.

Installation and out-of-the-box functionality is covered, followed by a more in-depth exploration of the implementation details. There are also some comments on how the standard offering can be customised for local needs.

Appendices provide command references and suggestions for further information.

Jane Curry
Skills 1st Ltd
2 Cedar Chase
Taplow
Maidenhead
SL6 0EU
01628 782565

jane.curry@skills-1st.co.uk

1 Overview

NetView 7.1.5 became available in October 2006 and, in collaboration with IBM Tivoli Monitoring (ITM) 6.1 Fixpack 3, it provides the ITM Tivoli NetView Server Agent. This is an ITM agent that provides graphical and table-based views showing both the health of the NetView system and some indication of the network that NetView is managing.

1.1 NetView overview and terminology

NetView 7.1.5 is available for AIX, Sun, Linux and Windows platforms and all support the ITM Tivoli NetView Server Agent. The NetView Server discovers networks, segments, nodes and interfaces using ping initially and then the Simple Network Management protocol (SNMP) to obtain more information from a device, if that device supports SNMP. NetView stores information in a number of proprietary-format databases and arranges discovered elements according to the rules of TCPIP.

Once an interface or node has been added to NetView's database, by default it is ping-polled every 5 minutes. This status polling can be configured for different polling intervals, timeouts, retries and to use SNMP rather than ping, but fundamentally it is this status polling that determines whether an element is up (green), down (red) or marginal (yellow). Status propagates up from the interface level so a router with at least one down interface and at least one up interface, will be yellow, as will the network segment that contains the router, as will the network that contains the network segment.

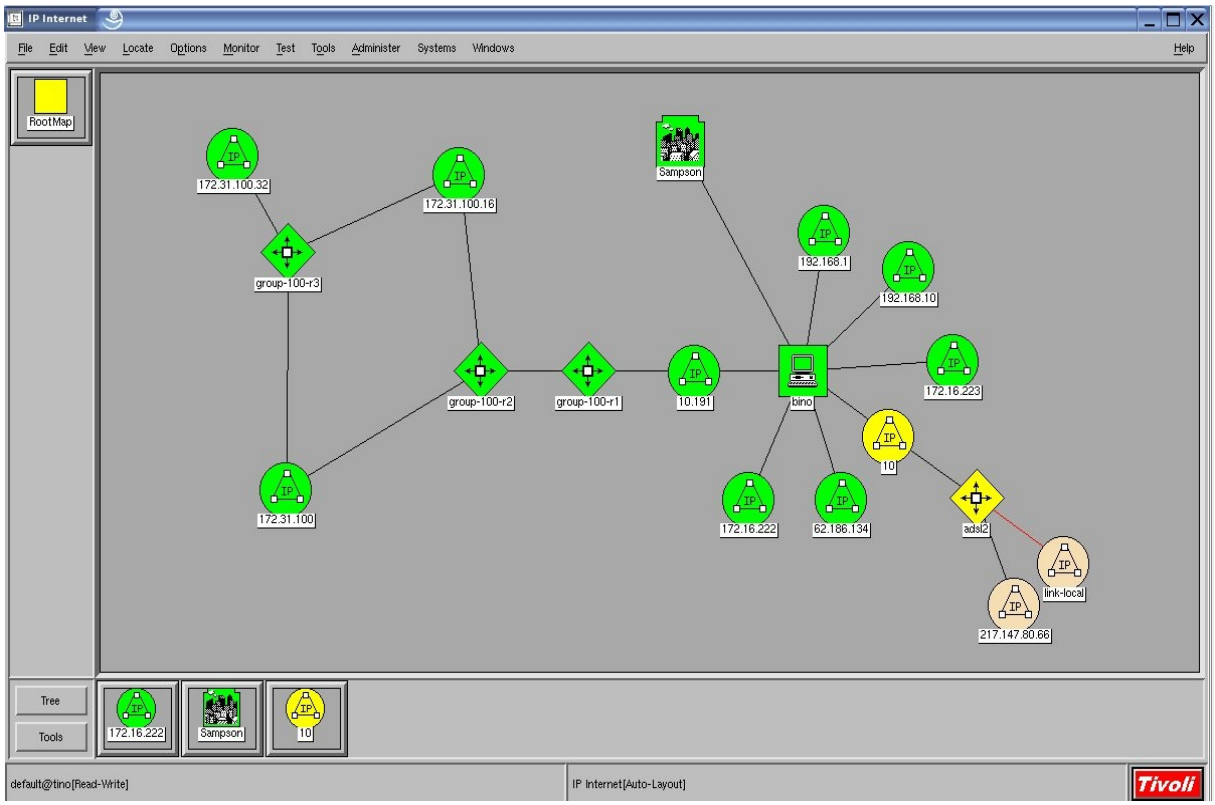

Figure 1: NetView IP topology

The default means of arranging elements discovered by NetView, follows the rules of TCPIP and the topology is shown under the "IP Internet" icon. The "SmartSets" icon provides a way to group elements based on other criteria – some default examples are "Routers" and "Cisco_Switches". NetView administrators have great flexibility to create their own SmartSet groupings.

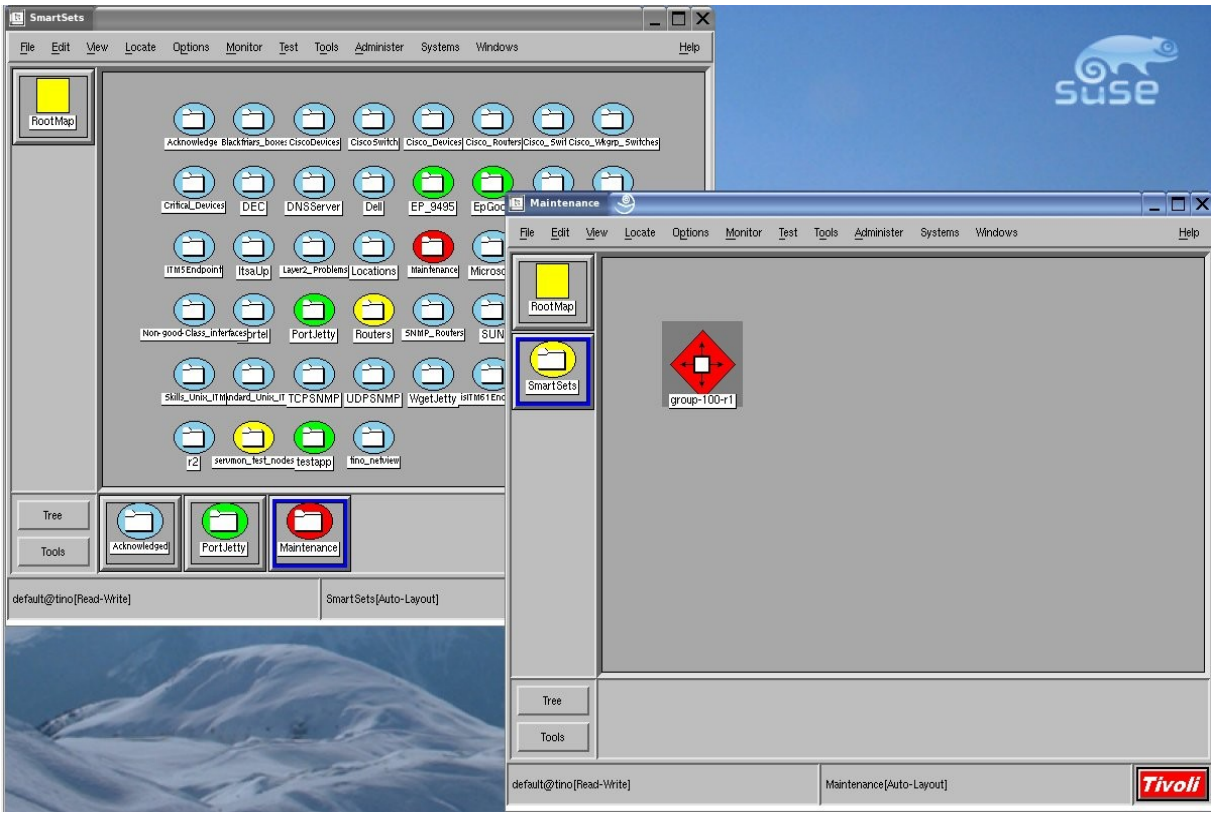

Figure 2: SmartSet submap showing Maintenance SmartSet opened

Events are displayed through the NetView Graphical User Interface (GUI). SNMP traps may be generated by the NetView Server (for example when it detects that a device did not respond to a status poll) or SNMP traps may be sent to the NetView Server from SNMP-enabled devices in the network. NetView has a comprehensive ability to process traps and respond to them either with local actions or by forwarding NetView events to a Tivoli Enterprise Console (TEC) Server.

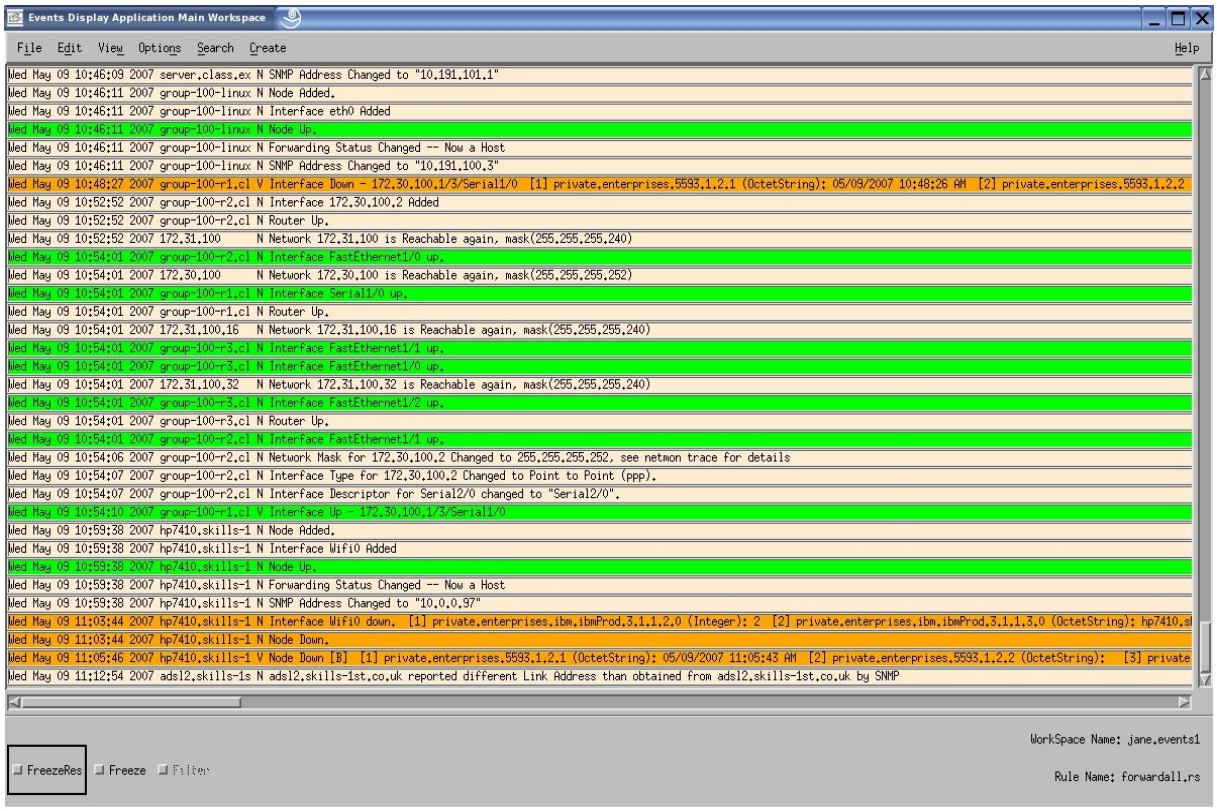

Figure 3: NetView Event Log

Although NetView itself has no inherent means of performing TCPIP name-to-address resolution, it relies very heavily on the underlying Operating System to perform this service quickly and accurately; indeed the more important function for NetView is to be able to resolve addresses to names, rather than vice versa.

1.2 ITM 6.1 overview and terminology

IBM Tivoli Monitoring (ITM) 6.1 was shipped in November 2005 with Fixpack 3 arriving in October 2006. The base ITM6.1 product delivers *systems* management using a number of operating system agents to deliver metrics such as CPU utilisation, disk space, memory usage, processes, services, etc.

Additional monitoring is available by deploying extra agents such as Database, Message Queueing, Unix logfile.

The focal point for an ITM 6.1 environment is a machine installed as a hub Tivoli Enterprise Monitoring Server (TEMS). For scalability, extra remote TEMS can be installed. ITM 6.1 agents are called Tivoli Enterprise Monitoring Agents (TEMAs) and are connected either to the hub TEMS or to a remote TEMS. The Graphical User Interface for ITM 6.1 is the Tivoli Enterprise Portal (TEP) which can either be a standalone Java Desktop (TEPD) or can run in an Internet Explorer 6 Browser. TEPs are connected through a Tivoli Enterprise Portal Server (TEPS).

Historical data can be gathered into a Tivoli Data Warehouse database via a Warehouse Proxy Agent; the Summarization and Pruning Agent can be configured to summarise data and prune old data.

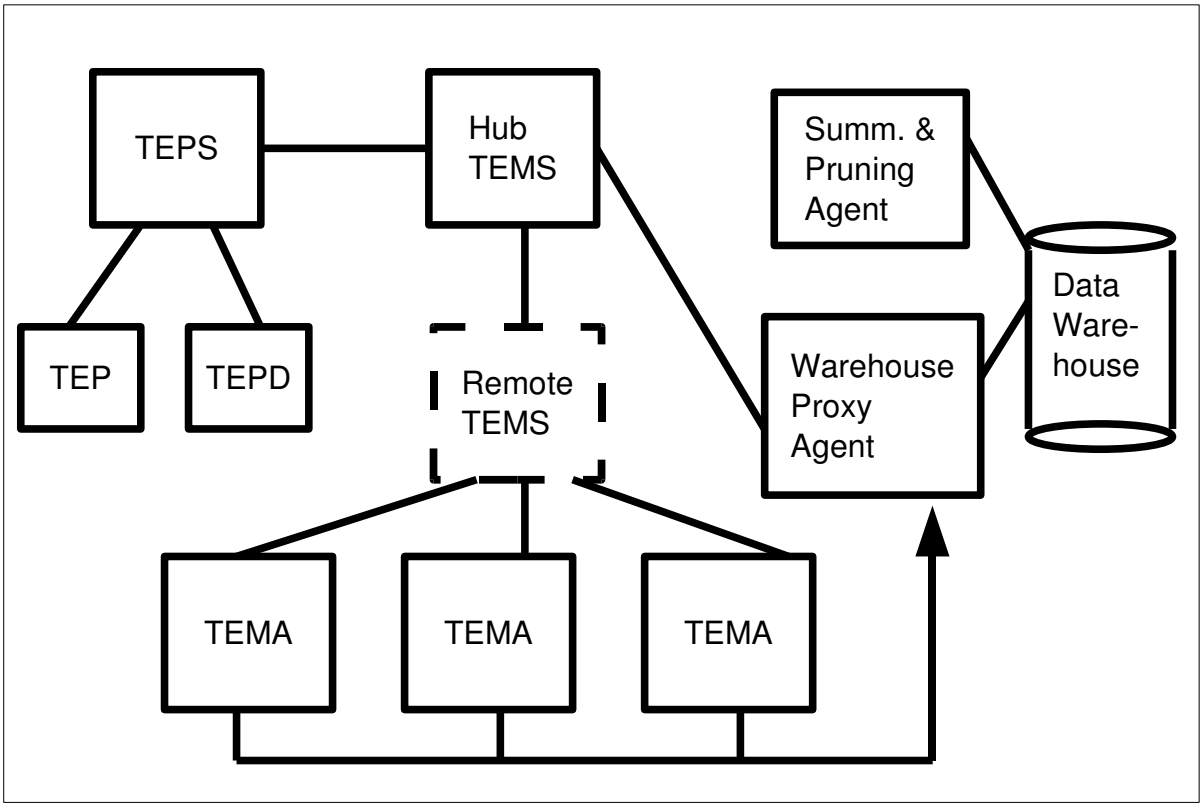

Figure 4: ITM 6.1 architecture

The ITM 6.1 TEP GUI usually displays a *Navigator* window to navigate around the different devices and functions being managed. Each Navigator item has associated with it, one or more *Workspaces*. Each Workspace may contain one or more *Views*, where a View is a window that may contain a table, graph, console, etc.

In addition to providing data attributes to populate graphical and table views, an ITM TEMA can also be configured to evaluate configured *situations*. Typically a situation is a threshold on a TEMA attribute that represents a problem; for example, CPU utilisation greater than 80% on 3 consecutive sampling intervals. Situations can also be configured with automatic response actions. Currently open situations are displayed in a *Situation Event Console* view.

As with NetView, ITM 6.1 has an option to forward events from ITM to a TEC Server. The TEC Java GUI is another view type in the ITM TEP console.

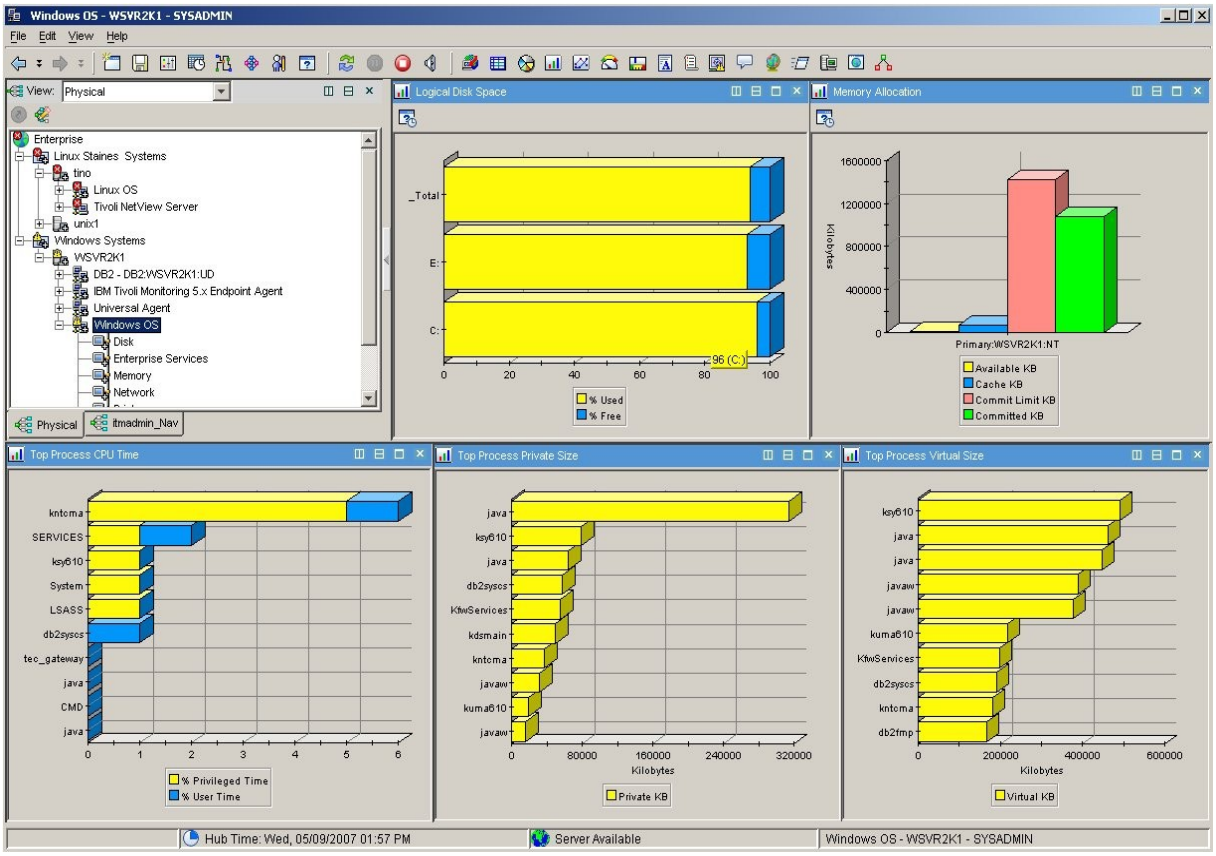

Figure 5: TEP showing Windows OS Navigator workspace which contains 5 views

1.3 ITM Tivoli NetView Server Agent overview and terminology

The ITM Tivoli NetView Server Agent is packaged in a similar manner to any standard ITM 6.1 TEMA. Once installed, it can provide data to populate a number of standard workspaces that are created automatically for each NetView TEMA. In addition, a number of ITM 6.1 situations are defined which can be distributed to any NetView TEMA. The standard views and situations address three areas:

- NetView Availability
- NetView Health
- Network Health

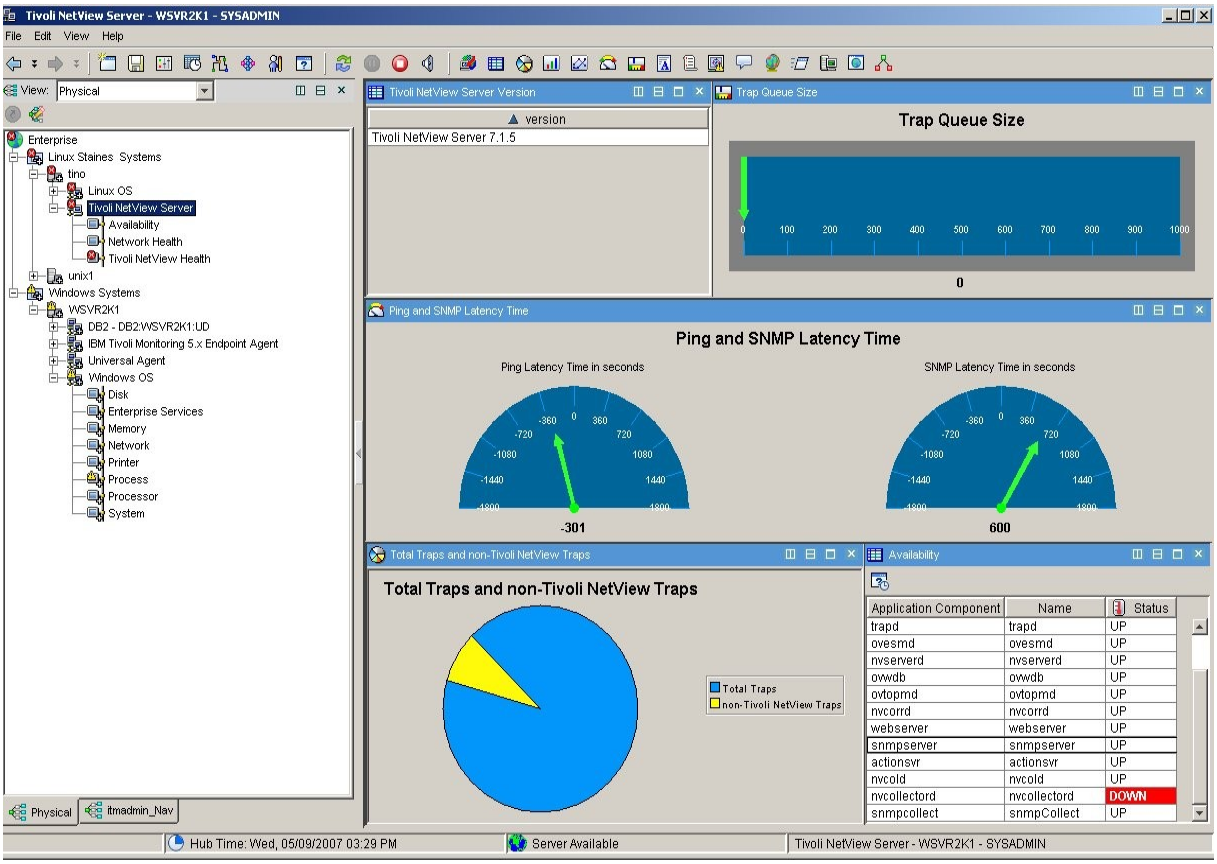

Figure 6: NetView TEMA - Tivoli NetView Server overview workspace

1.4 Other ITM 6.1 / NetView / TEC integration possibilities

As has already been mentioned, both NetView and ITM 6.1 can forward their events up to a TEC Server where further correlation and automation can be configured.

Not only can a TEC Java console be displayed as a view in an ITM 6.1 TEP Console, but also the standard functionality of selecting an event in a TEC Console and then calling up a NetView Web Console is perfectly possible from a TEC-in-a-TEP.

TEC rulesets are provided both for NetView / TEC integration and for ITM 6.1 / TEC integration.

NetView also offers the itmquery facility to provide a view of the ITM 6.1 architecture from a NetView perspective. Designed originally to provide integration between NetView and ITM 5.x, the itmquery function in NetView 7.1.5 has been extended to also report on ITM 6.1 agents.

2 Installation Details

The environment that I have used for this paper consists of:

- Windows 2000 Server (hostname wsvr2k1)
  - Hub TEMS, TEPS, Data Warehouse, Warehouse Proxy Agent, Summarization & Pruning Agent, TEPD, DB2 for TEPS and TDW, Windows TEMA
- Linux SuSE 9 (hostname tino)
  - Tivoli Management Region (TMR) Server, TEC Server, NetView Server, Linux TEMA, TEPD, DB2 for TEC and NetView

The ITM Tivoli NetView Server Agent is delivered on the “Options 2” CD of the set of NetView CDs. With Version 7.1.5, each NetView architecture now has its own set of CDs. ITM 6.1 Fixpack 3 or later is required to support the NetView TEMA.

2.1 Prerequisites

The prerequisites and supported architectures for the NetView TEMA are documented in the “Tivoli Monitoring: Tivoli NetView Server Agent Users Guide”, GC32-1859-00. Supported architectures include:

- AIX 5.1, 5.2, and 5.3 (32-bit and 64-bit)
- Sun Solaris 9 and 10 (64-bit)
- Red Hat Enterprise Linux 4.0 (Intel 32-bit)
- Red Hat Enterprise Linux 4.0 for zSeries (31-bit kernel on 64-bit hardware)
- SUSE Linux Enterprise Server 9 (32-bit)
- SUSE Linux Enterprise Server 9 for zSeries (31-bit kernel on 64-bit hardware)
- Windows 2003 Server EE (32-bit)
- Windows 2003 Server SE (32-bit)
- Windows 2003 Enterprise x64 Edition
- Windows 2003 Standard x64 Edition

Installation of the NetView TEMA requires at least 220Mb of free disk space for Unix variants and at least 160Mb for the Windows NetView TEMA.

In order to be able to communicate with the NetView TEMA remotely or through the TEP GUI, the Operating System TEMA for the appropriate NetView Server architecture, also needs to be installed and running.

The NetView TEMA User's Guide is rather light on installation details, often referring to the ITM 6.1 Installation Guide. Basically the process for

installing the NetView TEMA is similar to any other ITM 6.1 TEMA and requires installation of:

- The NetView TEMA itself on to the NetView Server
- ITM 6.1 Application Support for the NetView TEMA on all TEMS
- ITM 6.1 Application Support for the NetView TEMA on the TEPS
- ITM 6.1 Application Support for the NetView TEMA on all TEPD

Bear in mind when obtaining the "Options 2" CD that you may need support for different architectures. For example, if your NetView Server is Linux, your TEMS is Windows and your TEPS is AIX, you will need the NetView Options 2 CDs for all three different architectures!

Within each "Options 2" CD there are two subdirectories:

- ITM contains the TEMA and Application Support code
- IRA contains national language support (if required)

Every ITM 6.1 TEMA has a unique 2-character code to denote the TEMA type. The code for the NetView TEMA is **nd**. (Note that this 2-character code is sometimes prefaced by **k** so inspection of the code will show many files starting **knd** .

Typically, an ITM 6.1 TEMA can either be installed locally or remotely.

2.2 Local installation of a NetView TEMA

A local install of an ITM 6.1 TEMA is very straightforward with either a **setup.exe** for a Windows TEMA or an **install.sh** for Unix. For the NetView TEMA, remember to change into the **ITM** subdirectory first.

2.3 Remote installation of a NetView TEMA

In an ITM 6.1 paradigm, typically there are many targets that require the same TEMA – Operating System TEMAs for Unix, Linux and Windows are obvious examples. For these TEMAs, it makes sense to have a method for holding code in a central repository and providing a remote installation mechanism. Unless an organisation has several NetView Servers then there is less need for a remote installation mechanism. That said, the NetView TEMA **does** have utilities for remote installation; however, they are somewhat quirky, at least for the Unix versions of the NetView TEMA.

Remote installation of any ITM 6.1 TEMA involves copying the code from the distribution CD into an ITM 6.1 *Depot,* typically on the hub TEMS. This is done with a series of *tacmd* commands.

The first step is to authenticate yourself to the ITM 6.1 hub TEMS with a valid user name and, if required, the appropriate password. Note that ITM 6.1 will normally operate with security enabled at the hub TEMS which

implies that the user must be configured both to the hub TEMS Operating System and as a user in the TEPS. It is possible to operate with ITM 6.1 security **not** enabled at the hub TEMS in which case a user must simply be defined at the TEPS and no passwords are used. The default user that ships with ITM 6.1 is **sysadmin** and hub TEMS security is turned off at initial installation.

- tacmd login -s <TEMS> -u <user> -t 1440
- Eg. tacmd login -s wsvr2k1 -u sysadmin -t 1440
- This logs in to the TEMS for a session that will last 1440 minutes (1 day – if this parameter is not specified, the login session will last 20 minutes.). If the "-p <password>" parameters are not specified then the user will be prompted for the password. If passwords are not in use, simply hit Carriage-Return.

To see packages already available in the depot, use:

- tacmd viewDepot
- By default, the depot on a Windows TEMS will be in **C:\IBM\ITM\cms\Depot** and a Unix TEMS depot will be in **/opt/IBM/ITM/tables/<TEMS name>/depot** .

To import the NetView TEMA into the depot from a CD use the following:

- tacmd addBundles -i <CD>/ITM/unix -t nd
- This assumes a Unix Options 2 CD and you are importing the TEMA of type nd (ie NetView)
- tacmd addBundles -i <CD>\ITM\WINDOWS\Deploy -t nd
- This assumes a Windows Options 2 CD

Both architecture types have TEMA prerequisites that are also included in the same directory as the NetView TEMA. These prerequisites include Java, GUI prerequisites, the ITM product installer and shared libraries. Typically these prerequisites may well already be installed in a depot; otherwise they can be imported into the depot, as required. A "tacmd addBundles" command without a "-t" parameter will import all bundles available on the media.

Other useful tacmd commands for handling bundles are:

- tacmd listBundles -i <CD>/ITM/unix
- tacmd removeBundles -i <CD>/ITM/unix
- Note that the removeBundles command requires the "-i" parameter, and the appropriate CD, even though you are removing packages from the depot!

Once the depot contains the appropriate NetView TEMA code, it can, in theory, be deployed in two ways:

- From the TEP GUI, right mouse against the host in the Navigator that you wish to deploy the NetView TEMA to, and choose “Add Managed System”. You will be provided with a dialogue to select the NetView TEMA.
- tacmd addSystem -n <name of Operating System TEMA> -t nd
- Eg. tacmd addSystem -n tino:LZ -t nd

Unfortunately there seem to be a few glitches with the remote installation mechanism for the NetView TEMA!

My experience is that importing the Unix NetView TEMA using the “tacmd addBundles” command doesn't work (the Windows bundle is fine). There appears to be confusion in the installation files over the exact version of the NetView TEMA with references to:

- 071005000 in some files
- 071500000 in other files
- 081000000 in others!

In order to successfully import the bundle, I have made the following changes (Note that testing has only been done with the SLES 9 NetView TEMA where the TEMS and TEPS were both on a Windows platform):

- Copy the whole ITM subdirectory from CD to writeable media
- Determine the exact architecture of your NetView Server, eg. li6242
- Inside the ITM/unix subdirectory, modify the **.dsc** file that corresponds to the architecture for your NetView server eg. ndli6242.dsc
- You need to make changes in 4 places. Each one will reflect a version of 71500000:
  - <Release> line
  - <Mod> line
  - The last <File> line
  - The first <Command> line

With these changes consistently referencing version 7150000, the “tacmd addBundles” command can be used with the “-i” parameter pointing at your modified ../ITM/unix directory.

The next problem is that this image still does not install correctly using either “tacmd addSystem” or using the “Add Managed System” menu from the TEP. Both methods can apparently pick up the depot bundle but they

both fail in an identical manner complaining about either disk space or insufficient authorisation (I have tested this whilst authenticated as sysadmin). If you try to deploy an agent remotely and get "An agent configuration schema was not found", this probably means that you have not installed the Application Support on your TEMS and TEPS (see next section).

Although detailed testing has only been carried out with the SLES 9 NetView TEMA from a Windows TEMS and TEPS, I have also tried the same procedure from a SLES TEMS depot and see exactly the same results on both "tacmd addBundles" and the same subsequent failure on remote deployment. This is currently the subject of a PMR.

I have not installed other Unix variants of the NetView TEMA but inspection of the Solaris TEMA code shows the same confusions over version numbers.

2.4 Post TEMA installation tasks

Once the NetView TEMA code has been deployed, the agent needs to be configured.

The Tivoli NetView Server Agent Users Guide suggests configuring a Windows NetView TEMA using the graphical "Manage Tivoli Monitoring Services" application (this should have been installed automatically with the NetView TEMA if it was not already present on the box – it is the ITM 6.1 "traffic light" icon). This application should show the NetView TEMA but it will not be running. Use the right mouse button to select the "Reconfigure" menu. You will need to specify the hostname of your TEMS correctly – otherwise take defaults unless you know otherwise for your environment. To start the NetView TEMA, double-click the agent – the icon beside the TEMA should change to a running blue man. (Be aware that the status of TEMAs in the "traffic light" application may sometimes need Refreshing from the View menu, to update recent status changes).

The User Guide suggests configuring a Unix NetView TEMA using the "itmcmd" command though it is perfectly possible to use the "Manage Tivoli Monitoring Services" application, as documented for the Windows variant. To start this application use "itmcmd manage". Note that when working on Unix ITM 6.1 systems, it is advisable to ensure that the **CANDLEHOME** environment variable is set (/opt/IBM/ITM is the default) and that the PATH environment variable includes **$CANDLEHOME/bin** . The "traffic light" application behaves slightly differently under Unix in that some menus are missing and double-clicking on an agent does not start/stop it – you can either use the right mouse menu to access Start / Stop or click on the red / green traffic light icon having first selected the appropriate TEMA.

To configure the NetView TEMA using "itmcmd", use:

- itmcmd config -A nd and supply the correct hostname for TEMS

- Note that the User's Guide misses “config” from the above command!

The User's Guide suggests starting and stopping the NetView TEMA using:

- CandleAgent start nd

I have had very variable results with this, including being unable to stop/start the NetView TEMA without completely bouncing the Operating System TEMA first (the error message in the logfile was “Could not find configuration file”). I have had similar results using tacmd to stop, start or restart the agent:

- tacmd stopAgent | startAgent | restartAgent -t nd

Provided the NetView TEMA is installed on a box that also has the appropriate Operating System TEMA, it should be possible to Start, Stop, Configure, Restart and Remove the NetView TEMA using the right mouse menu from the agent in the TEP GUI. Again, I have had variable success with these – Start / Stop sometimes works but I have had no joy with Configure or Remove.

The most reliable method of controlling the NetView TEMA seems to be the “itmcmd” command:

- itmcmd agent stop | start nd

The “traffic light” “Manage Tivoli Monitoring Services” GUI seems to be the next most reliable.

Note that both tacmd and itmcmd are *local* commands; the TEP is the only easy way to control remote agents.

Logfiles for the Unix NetView TEMA are in $CANDLEHOME/logs with filenames such as <hostname>_nd_<hex timestamp>.log . Note that this is slightly different from what the manual says! A new logfile is created whenever the TEMA is restarted.

2.5 Installing Application Support for the NetView TEMA

Application Support for the NetView TEMA is required for the TEMS, TEPS and for any Desktop TEP GUIs (TEPD). Installing Application Support used to be known as *seeding* and is still referred to that way in some ITM 6.1 documentation. If you forget to install the TEMS support, you will not see any of the NetView TEMA situations or the NetView Managed System List. If you don't install the TEPS or TEPD support then menus in the TEP will not be correct and Navigator items will have strange names.

The Application Support code for TEMS, TEPS and TEPD comes on the same NetView Options 2 CD as the TEMA – do remember that you may need several Options 2 CDs if your TEMS / TEPS / TEPD architectures are different and differ from the NetView Server architecture.

To install Windows versions of the Application Support, simply run the **setup.exe** in <CD directory>\ITM\WINDOWS . Choose to install TEMS, TEPS and TEPD Application Support, as required on this system; do *not* install the Agent unless you have not already installed the NetView TEMA and wish to do so on this system. You will be prompted through the installation panels – the bits you need to get right are the hostnames of your TEMS and TEPS. Any active TEP GUIs will need restarting after Application Support installation. There are no further tasks required to initialise the Application Support on a Windows system.

To install Application Support for the NetView TEMA on Unix ITM 6.1 infrastructure, change to the <CD directory>/ITM directory and run the **install.sh** script. Choose to "Install products to the local host". Take defaults until you reach the "Operating Systems" question. At this point, choose:

- 12 for TEMS Application Support
- 11 for TEPS Application Support
- 10 for TEPD Application Support
- 9 for TEP Browser Application Support

You will be prompted again for the Application Support you want to install. It should show the NetView TEMA as the only selection so you can take the default "all of the above". The Application Support installation dialogues seem rather confusing as you select to install Application Support and you are then prompted with what looks like a TEMA install – it's only when you have accepted it that a comment shows that it is *Application Support for the NetView TEMA* that you are installing, not the TEMA itself.

For Unix Application Support there are a number of post-installation tasks required:

- After installing TEMS Application Support, run:
  - itmcmd support -t <TEMS> nd
  - For example, itmcmd support -t HUB_tino nd
  - The <TEMS> is the name of the TEMS Managed System, not the hostname of the box on which the TEMS is installed
  - Stop and start the TEMS either using the "Manage Tivoli Monitoring Services" GUI or "itmcmd server start|stop <TEMS>"
- After installing TEPS Application Support, run:
  - itmcmd agent stop cq
  - itmcmd config -A cq
  - itmcmd agent start cq

    - (cq is the 2-character code for TEPS)
- After installing TEPD Application Support, run:
    - itmcmd config -A cj
    - (cj is the 2-character code for TEPD)

2.6 Uninstalling the NetView TEMA

To uninstall the NetView TEMA from a Windows system, use the standard Windows technique of "Add / Remove Programs" from the Control Panel. Select the ITM 6.1 Program and take care to select **Modify**. Only select **Remove** if you want to totally remove *all* elements of ITM 6.1 from this system. The "Modify" selection should provide a list of installed ITM 6.1 elements from which you can choose which to remove.

To uninstall the NetView TEMA from a Unix system, change to $CANDLEHOME/bin and run the **uninstall.sh** script. You will be presented with a list of installed elements from which you can choose which to remove.

3 Out-of-the-box functionality

The ITM Tivoli NetView Server Agent, along with its Application Support, provides a number of standard Workspaces and Views for the TEP GUI, some Situation definitions to display NetView-related problems in the TEP Situation Event Console, and BAROC and rules files to help integrate NetView TEMA situation events into a Tivoli Enterprise Console solution.

Most of the NetView TEMA management is implemented by sampling *attributes* that are grouped into *attribute groups*. The Tivoli NetView Server Agent Users Guide, Chapter 5, documents all the attributes and attribute groups instrumented by the NetView TEMA.

3.1 Workspace views

For any NetView TEMA that has been installed and configured, the agent should automatically appear in the TEP Physical Navigator as a **Tivoli NetView Server** , which has three sub-entries:

-
- Tivoli NetView Server — *views from other workspaces for overview*
    - Availability — *state of NetView daemons – memory, CPU*
    - Network Health — *state of nodes, i/fs, routers, networks,...*
    - Tivoli NetView Health — *traps, DNS response, ping/SNMP latency,.*

Each Navigator entry has one workspace associated with it. The Workspaces are refreshed by a user on-demand, not on a periodic basis.

Any view (window) within a workspace can always be made full-screen and restored using the window maximise / minimise buttons. This is particularly useful for bar graphs showing many variables.

3.1.1 Tivoli NetView Server workspace

The top-level Navigator item for a NetView TEMA contains views provide an overview of the state of the TEMA. The only view unique to this workspace shows the version of the NetView TEMA.

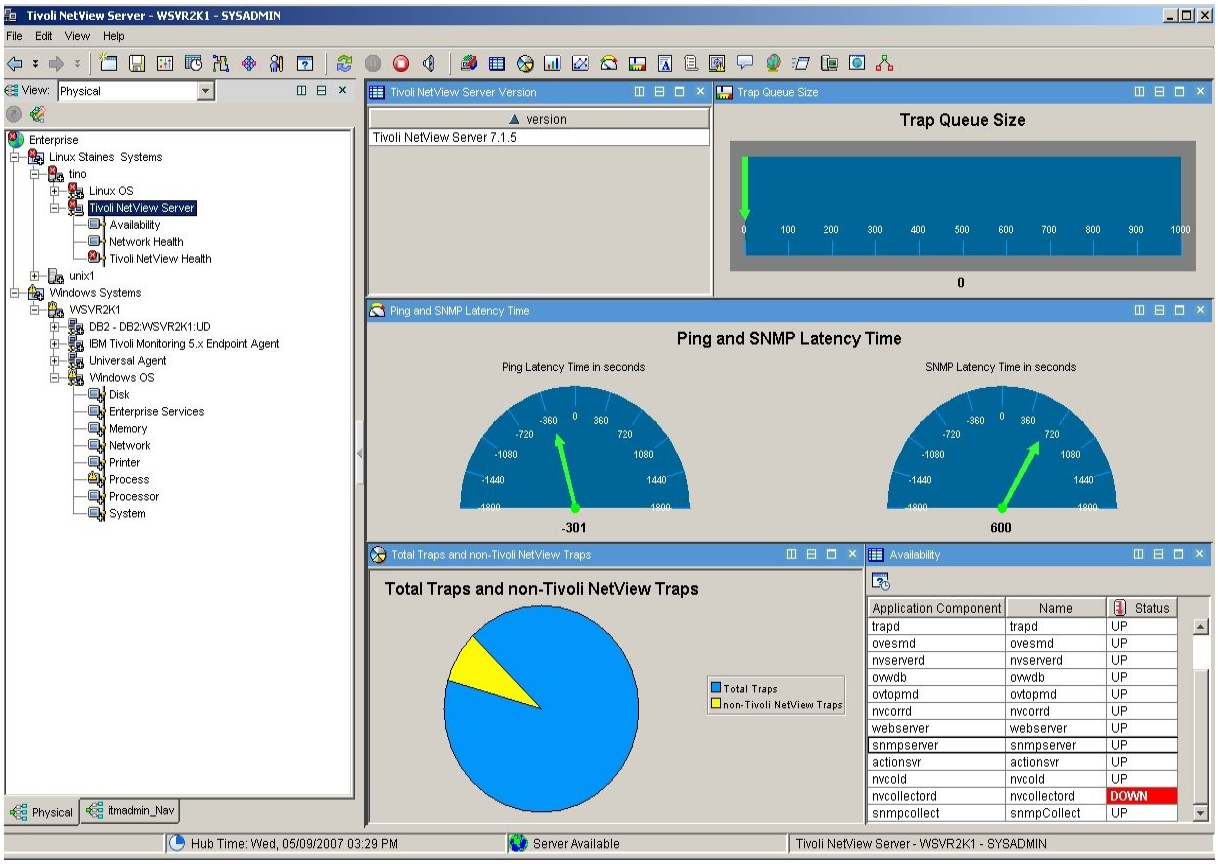

Figure 7: Tivoli NetView Server summary workspace with 5 views

3.1.2 Availability workspace

The Availability workspace uses a built-in subset of Operating System TEMA functionality to get information specifically on NetView processes:

- Whether processes are running
- Amount of CPU per process
- Memory used per process

- Threads per process (note that some Unix systems effectively “clone” a process to implement threads so no process will ever report more than one thread)

As shipped, the Availability workspace will always show at least one NetView daemon as DOWN. This is because NetView 7.1.5 now has two methods (and two separate daemons) for performing historical SNMP data collection. The original method uses **snmpCollect**; the new daemon with 7.1.5 is **nvcollectord** . These two daemons are mutually exclusive in NetView but both are included in the availability process view.

Chapter 4, page 13 of the ITM NetView Agent User Guide provides good documentation for changing the filters of any view. To access the Properties of any view, simply right-click in the view. The **Filters** tab allows you to prevent display of unwanted data (such as nvcollectord) or to include the display of other daemons. The “Data Snapshot” window at the bottom of the Properties page shows all the data that is actually retrieved from the NetView TEMA – any of the daemons shown in this window can be filtered in to the display simply by scrolling to the next empty cell in the “Application Component” column, clicking in it, and type the daemon name. The daemon name should be the same as the name used in the daemon's local registration file (daemon_name .lrf) in the Tivoli NetView product.

To prevent display of an unwanted daemon, select the row that contains its name in the “Application Component” (left click on the appropriate index number in the first column) and delete it using right-click and “Delete”.

Once the workspace has been modified to match your NetView environment, you should save that workspace with a different name (**File -> Save As** menu and provide a unique name). You may also want to make it the default workspace for that Navigator item (**Edit -> Properties** menu and click the “Assign as default for this Navigator Item” button).

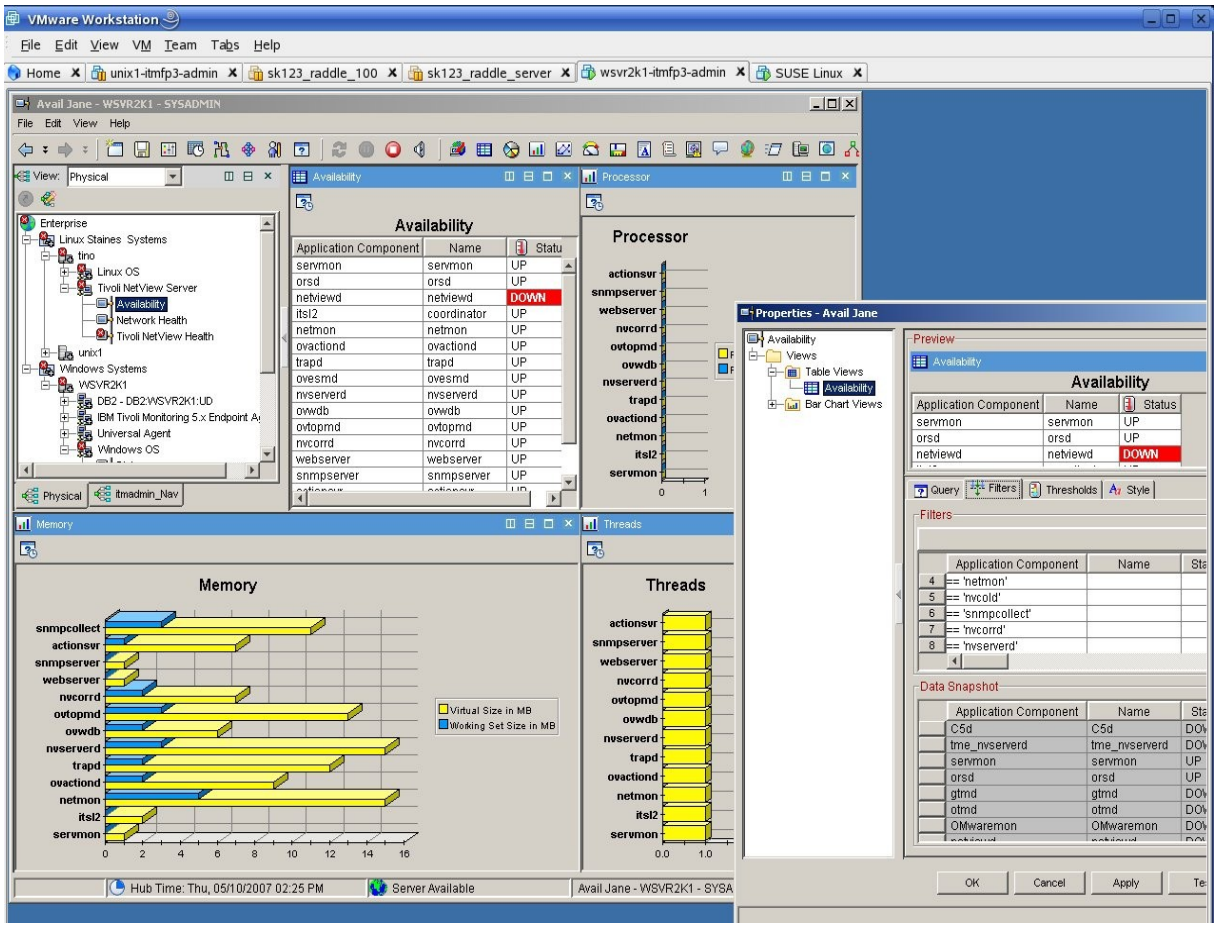

Figure 8: "Avail Jane" modified workspace with filtered process list

The "Availability" view within the "Availability" workspace (ie the process list) is also included in the "Tivoli NetView Server" overview workspace. You will need to change the filters for the view in the "Tivoli NetView Server" workspace as well as in the Availability workspace.

3.1.3 Network Health workspace

The Network Health workspace provides overview data collected from the NetView topology database. It includes four views representing:

- Node status – the number of nodes with each NetView status
  - The 8 NetView statuses displayed are Critical (red), Marginal (yellow), Normal (green), Unknown (light blue), Unmanaged (buff), Unreachable (white), User1 (pink) and User2 (purple).
- Nodes Up, Down, Discovered and Deleted
- Network elements - the number of different types of element – networks, segments, nodes, interfaces and routers
- SmartSet Membership – the number of elements in each SmartSet

Chapter 4 of this paper discusses in more detail how these values might be generated. What is not clear at first sight is which values are *snapshots of*

the current values in the NetView databases and which are *changes in values* since the last query of an attribute by the NetView TEMA. All the values in the Network Health workspace are snapshot values *except* the numbers of Nodes Deleted and Nodes Discovered - these two values are deltas since the last query by the TEMA.

The SmartSet Membership view may well be information overload if you use a fair number of SmartSets. Using the Filters tab of the Properties page for this view, it may be advisable to limit which SmartSets have totals displayed.

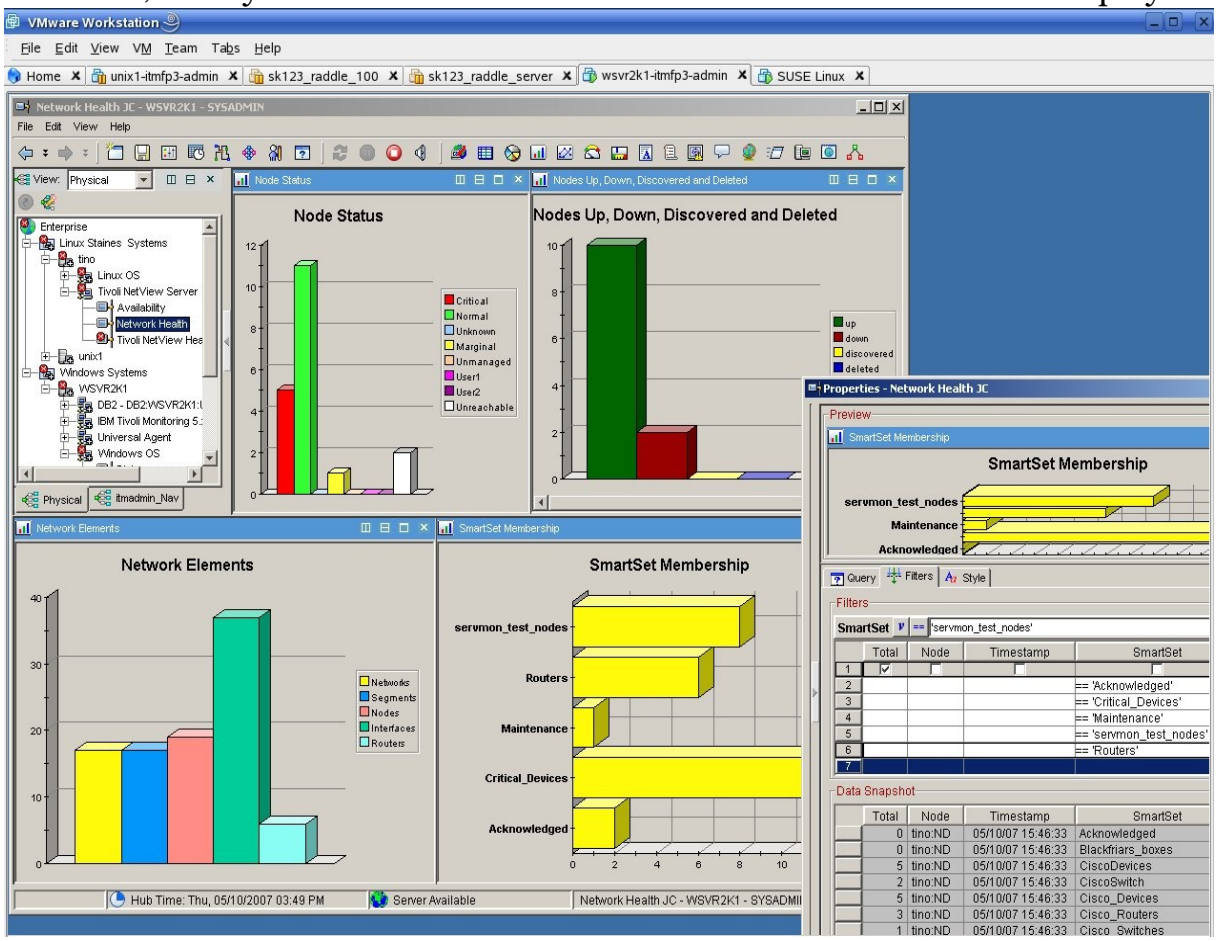

Figure 9: Network Health JC workspace with SmartSet Membership view customised

3.1.4 Tivoli NetView Health workspace

The Tivoli NetView Health workspace contains five views, as shipped:

- Total Traps And non-Tivoli NetView Traps
- Total Web Consoles
- DNS Response Time
- Trap Queue Size
- Ping and SNMP Latency

There are a number of details about this workspace that can be improved upon.

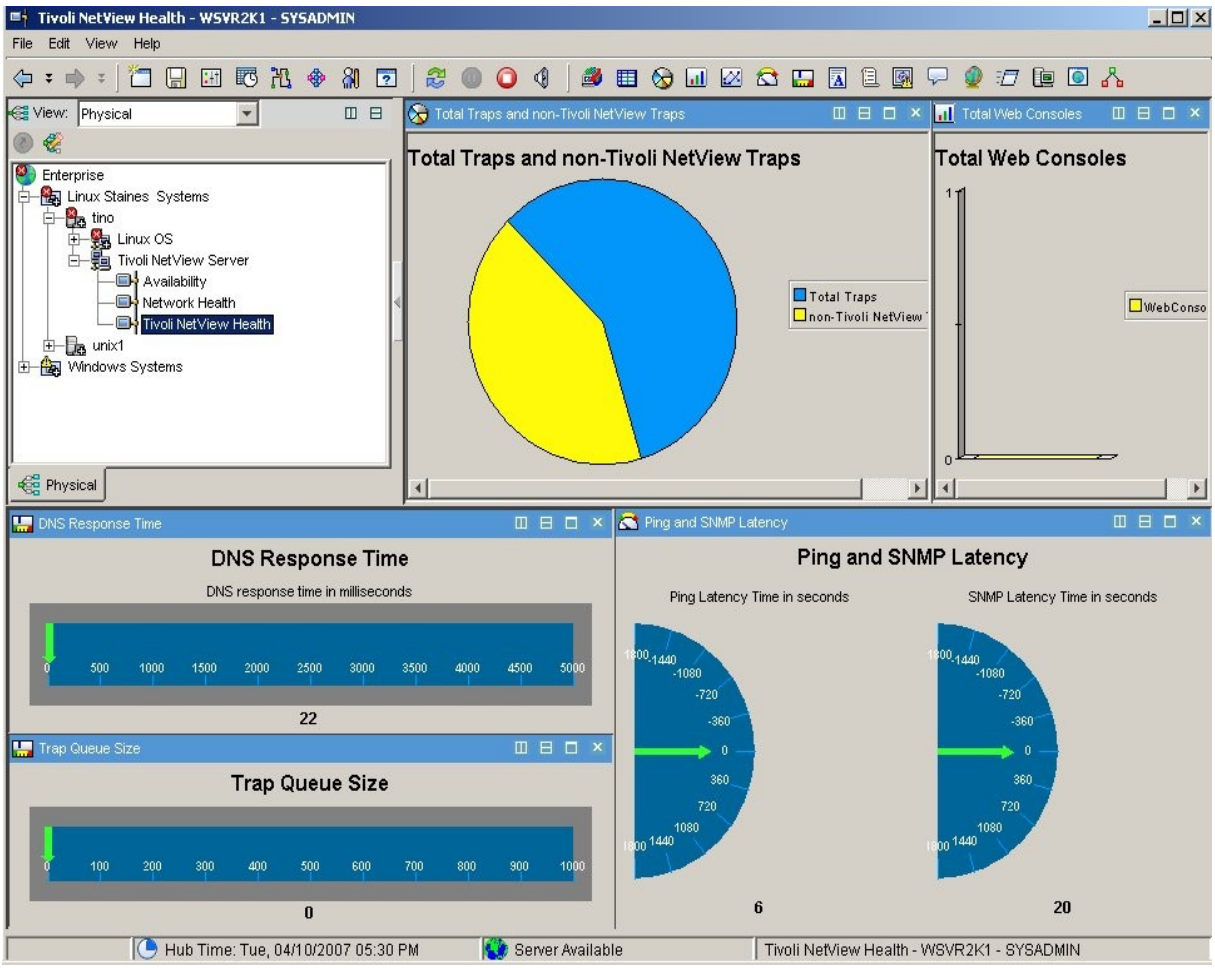

Figure 10: Default Tivoli NetView Health workspace

The "Total Traps and non-Tivoli NetView Traps" view shows delta values since the last time the NetView TEMA requested these values. non-Tivoli NetView Traps are those that do *not* come from the netView6000 enterprise (. 1.3.6.1.4.1.2.6.3.1) , ie. those that are generated by agents out in the network. non-Tivoli NetView Traps also include those generated by IBM Tivoli Switch Analyzer (ITSA).

The main issue with this view is that three Tivoli NetView Traps are generated in order to query for these attributes! They are:

- netmon-related Application connected to trapd
- Netmon Action
- netmon-related Application disconnecting from trapd

By default, these NetView internal traps do not display in a NetView Event Log and will not be forwarded to a TEC Server, but they may significantly distort the data in this ITM view, especially as you may have view refreshes, situations and ITM historical data collection all independently requesting the same data and generating trap triplets. This is logged as APAR IY97203 and has a temporary fix available which will be included with NetView 7.1.5 FP0001.

The use of a Pie Chart for this view seems rather strange – or rather the data is not suited to the Pie Chart. It would be far preferable to have the Pie Chart show "non-Tivoli NetView Traps" and "Tivoli NetView Traps", making up 100% of total traps. Unfortunately, the NetView TEMA does not provide a value for "Tivoli NetView Traps" and a workspace view cannot use data that is constructed from a formula. This view would benefit from an enhancement request.

One way to address this issue without requiring code changes to the NetView TEMA would be to change the view type to a bar chart showing "Total Traps" and "non-Tivoli NetView Traps" . This is done by:

- Click the Bar Chart icon from the menu of view icons at the top of the TEP
- Click in the pie chart view to change
- You will be presented with a screen that has the correct Query selected so click the "Select All" button to select the "Total Traps" and "non NetView Traps" attributes
- You will need to restore the title to the bar chart so right click on the bar chart and select the "Properties" menu
- Click the "Style" tab
- The header area should already be selected in the preview screen on the left so simply type the title into the text box. You will also need to click the "Show" box under the "Options" area.

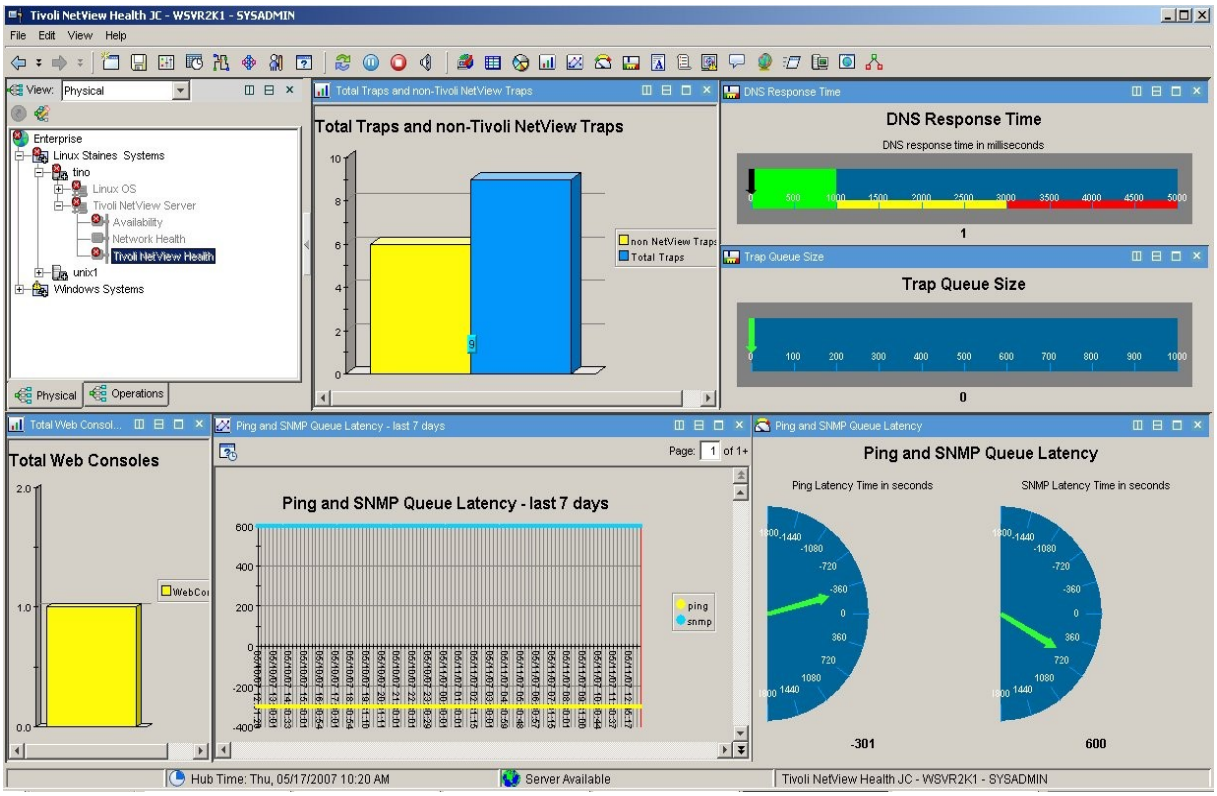

Figure 11: NetView Health workspace with bar chart rather than pie chart for Traps

A small but confusing issue with this view is that on random occasions, blue is used to show "non-Tivoli NetView Traps" and yellow is used for "Total Traps", rather than vice versa. Sometimes simply refreshing the workspace reverts the colours back to their default behaviour.

The Trap Queue Size view shows the number of traps that have been received by NetView but not yet been processed by NetView's trapd daemon. This should normally be zero unless a trap storm is in progress.

The "DNS Response Time" view should show DNS response time (in milliseconds) but it always shows the same value! The NetView TEMA uses the NetView /usr/OV/bin/nametest executable to deliver values (more details in Chapter 4). APAR IY97267 addresses two issues with nametest – a temporary fix is available which will be included with NetView 7.1.5 FP0001. This APAR does *not* yet address the problem that "good" values are returned even if NetView has no access to a DNS resolver!

The "Ping and SNMP Latency" view is rather misleading and is documented in Technote 1258387. Fundamentally, NetView maintains two internal queues, one for ping and one for SNMP, which determines when each node or interface will next be ping-polled or SNMP-polled. Each queue shows the number of seconds to the next poll. The "Ping and SNMP Latency" view shows the number of seconds until the next ping / SNMP poll (ie the

"top" entry for each queue). Hence the data is showing Ping / SNMP **Queue** Latency, *not* the actual time taken by a ping or an SNMP request.

This is why the gauges in the view (and indeed the supplied Situations) show and test for negative values. A negative value simply means that NetView is currently too busy to honour the queue schedules and is behind. This is something that may well happen in large networks and is not necessarily a critical problem provided the situation does not persist. It is, however, very useful to have a graphical display showing negative queue latency.

The title of this view can easily be changed by right-clicking and bringing up the Properties page. Use the "Style" tab and click on the Header area in the diagram to see and modify the title. Don't forget to save the workspace under a different name and to make your new workspace the default for the Tivoli NetView Health Navigator item.

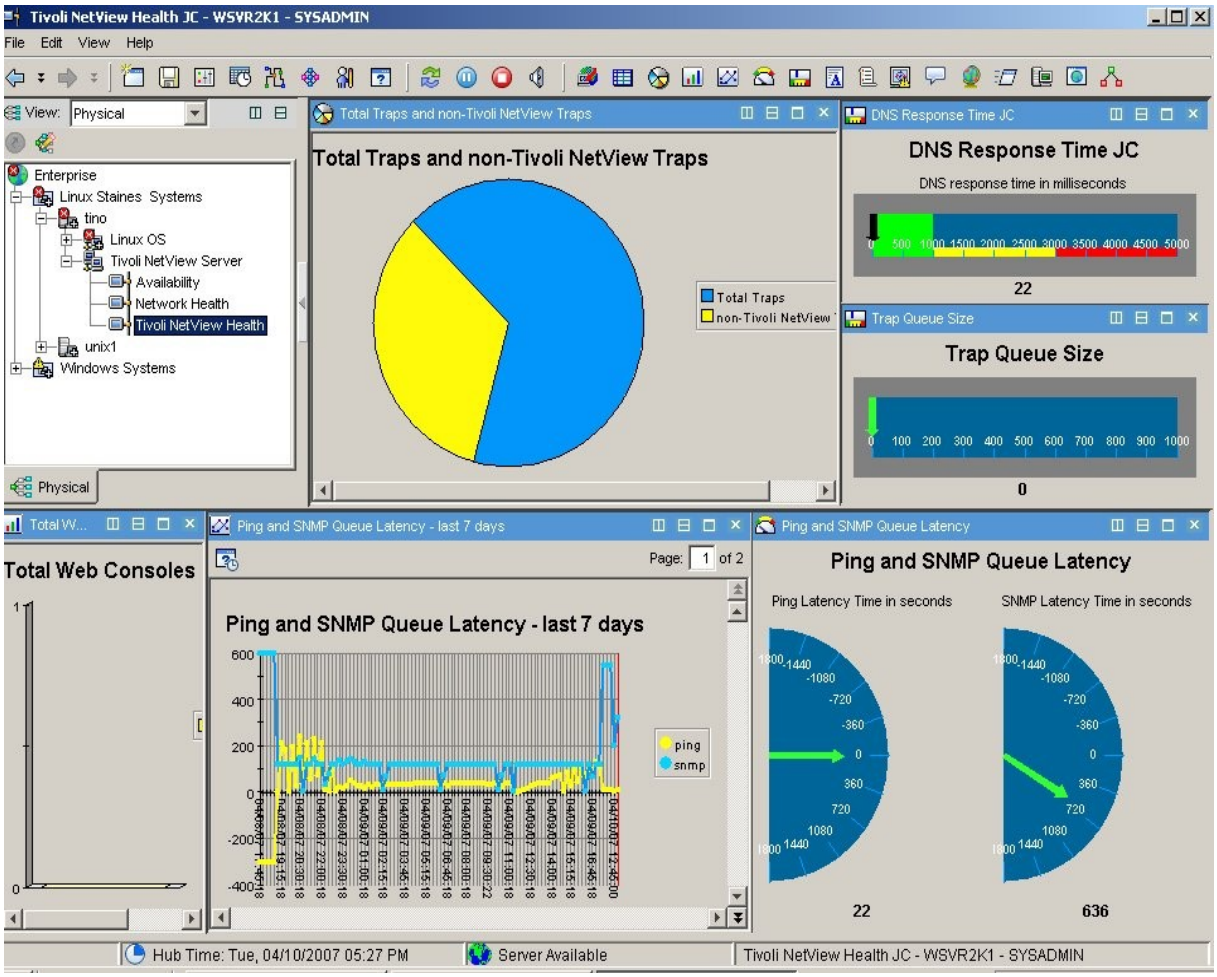

Figure 12: Modified Tivoli NetView Health JC workspace

There may be many other view customisations you wish to perform using standard ITM 6.1 configuration techniques. Figure 12 shows the inclusion of a view showing historical data for Ping and SNMP Queue Latency over the

last 7 days. Subranges have also been configured for DNS Response Time. This workspace may also be a good candidate for setting up an automatic refresh interval (**View -> Refresh Every** menu), rather than making it user demand-driven.

3.2 Situations

Situation definitions are the ITM 6.1 mechanism for defining a change – this is usually a "bad news" event (such as a NetView daemon is not running that should be). Most situations and all product-provided NetView TEMA situations, are *sampled*; that is, a TEMA attribute is polled on an interval basis and tested against a configured threshold. *Open situations* are displayed in a TEP Situation Event Console view. An icon should also appear against a Navigator item for which an open situation exists – the icon colour denotes the severity of Critical (red), Warning (yellow) or Informational (blue).

It is not normally possible to close sampled events manually from a TEP. The event will close automatically when the error condition clears. (Having said that, I have sometimes seen a "Close Event" menu option against NetView events.....).

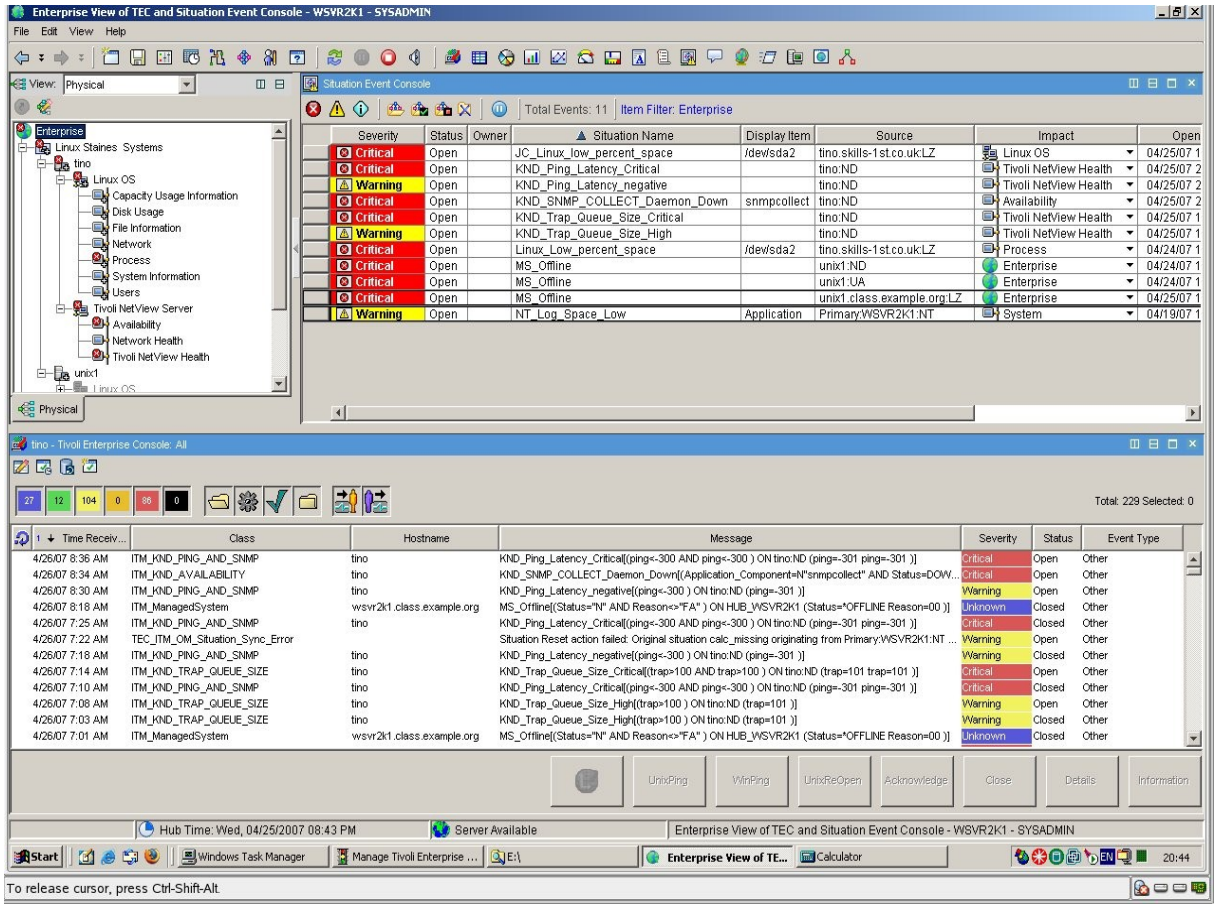

Figure 13: Situation Event Console for the entire Enterprise in top view - also note icons against Navigator items

All the product-provided situations for the NetView TEMA are documented in the NetView Server Agent User Guide, Chapter 6. All situation names are prefixed with KND; for example, KND_Ping_Latency_negative .

3.2.1 Availability associated situations

There are 22 pre-defined situations relating to NetView processes that are associated with the NetView TEMA Navigator item “Availability”:

- Separate situations for most NetView daemons down
- Situation for when process data is unavailable to the TEMA
- Situation for non-core daemons down (netviewd and nvpagerd)
- Situation for Unix non-core daemons down (ovelmd, pmd, tme_nvserverd)
- Situation for Windows non-core daemons down (nvcord, tecad)

By default, the standard NetView TEMA situations will be distributed to the ITM 6.1 built-in Managed System List called ***NETVIEW_SERVER_AGENT**. This list is built and maintained automatically as new NetView TEMAs are installed and / or removed. This means that all the NetView situations will

be distributed to all the NetView TEMAs (they only fire if a NetView daemon exists *and* the status is DOWN, so, for example, the Windows non-core daemons situation will *not* fire on a Unix NetView Server as the Windows-specific daemons will not exist).

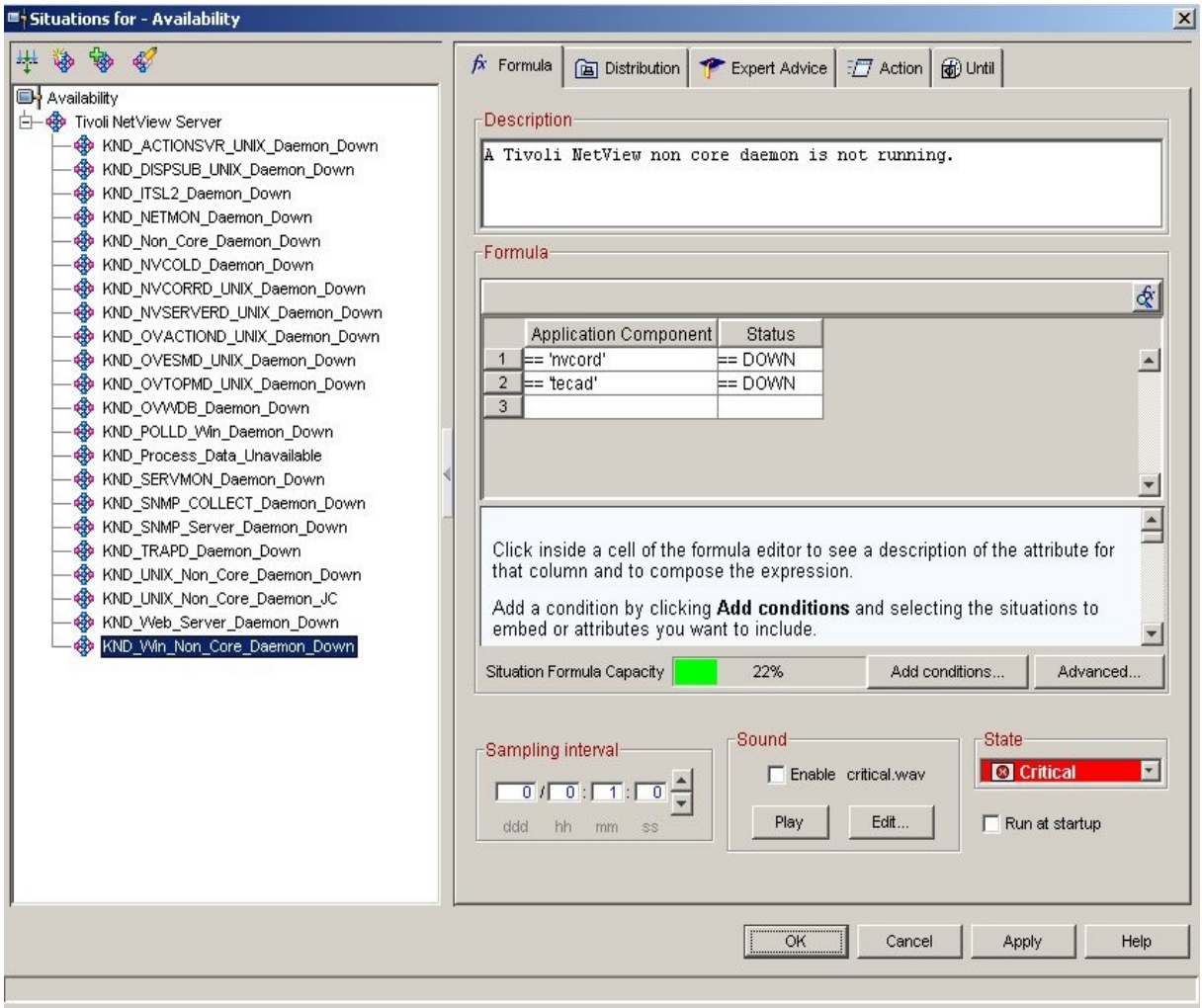

Figure 14: NetView TEMA situations to monitor NetView daemons

The process situations generally have “Take action = No”, “Run at Startup=Yes”, “Sampling Interval = 1 min”, “Severity = Critical” and “Situation persistence = 1” (the latter is behind the “Advanced” button). The exceptions to this are:

- The KND_Process_Data_Unavailable situation has a persistence of 3 and a severity of Informational
- All three Non_Core situations, KND_ITSL2_Daemon_Down and KND_SERVMON_Daemon_Down have “Run at Startup” set to “No”

Note that you will probably want to disable either the KND_NVCOLLECTORD_Daemon_down or the KND_SNMP_COLLECT_Daemon_down situation (as both cannot run at once

on a NetView Server).

Note that the KND_Win_Non_Core_Daemon_Down situation will not work for the NetView on Windows TEC adapter as the adapter is actually called tecad_nv6k, **not** tecad.

3.2.2 Tivoli NetView Health associated situations

There are 8 pre-defined situations relating to NetView health that are associated with the NetView TEMA Navigator item “Tivoli NetView Health”:

- Ping latency Warning / Critical (Warning at < -300s)
- SNMP latency Warning / Critical (Warning at < -300s)
- Trap queue size high / Critical (high Warning at > 100
- DNS response high / Critical (high Warning at > 3000ms)

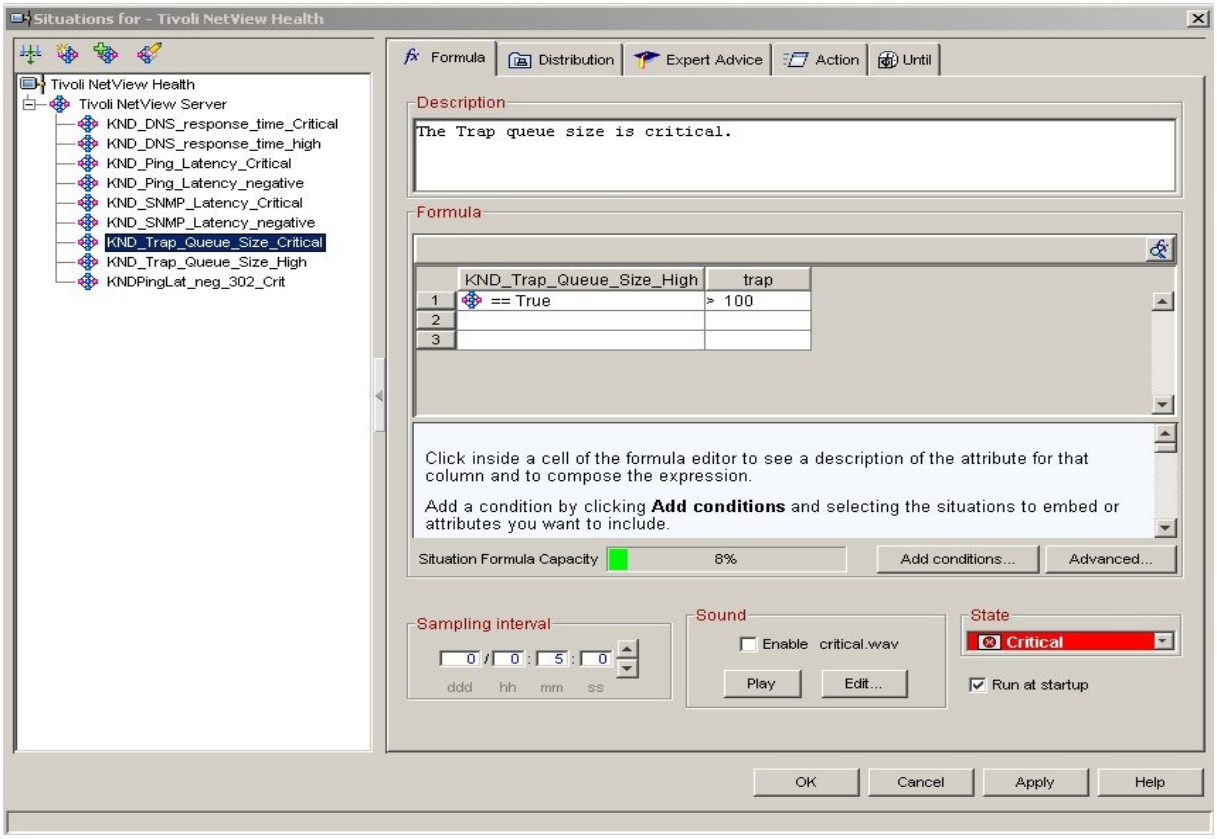

Figure 15: NetView TEMA situations monitoring NetView health

The NetView Health situations have “Take action = No”, “Run at Startup = Yes” and “Sampling Interval = 5min”. The warning events have a “Situation persistence = 1” with a Warning severity, whereas the critical situations have a persistence of 2 and a Critical severity.

Note that although KND_SNMP_Latency_negative and KND_Ping_Latency_negative are documented as having a Warning severity, they are actually shipped with a Critical severity.

You should not modify product-shipped situations (though it is possible). It is much better practise to copy a product-shipped situation and then modify it. Don't forget to Stop the product- shipped one, and to distribute your copy, and associate it with the appropriate node in the Navigator. To ensure that situations are correctly associated (and hence display properly), **always** start the Situation Editor by using the right-mouse menu option from the appropriate item in the Navigator.

All ITM Situation events map to TEC events and are forwarded to TEC.

3.3 TEC integration

Several out-of-the-box features are shipped to provide integration between TEC 3.9, ITM 6.1, NetView 7.1.5 and the ITM Tivoli NetView Server Agent.

3.3.1 Console integration

For many years it has been possible to start the NetView Web Console by selecting an event in a TEC Java Console and using the TEC NetView menus to start:

- A NetView Submap Explorer focused on the node referenced by the TEC event
- A NetView Diagnostics menu, similarly focused
- A NetView Object Properties menu, similarly focused

Note that the machine running the TEC Java Console must also have the NetView Web Console installed (although the TEC Console does **not** have to be a TME version, as some documentation suggests).

With ITM 6.1 it is now possible to run a TEC Console as a TEP view and the NetView submenus work in exactly the same way by selecting a TEC event and using a right-click menu.

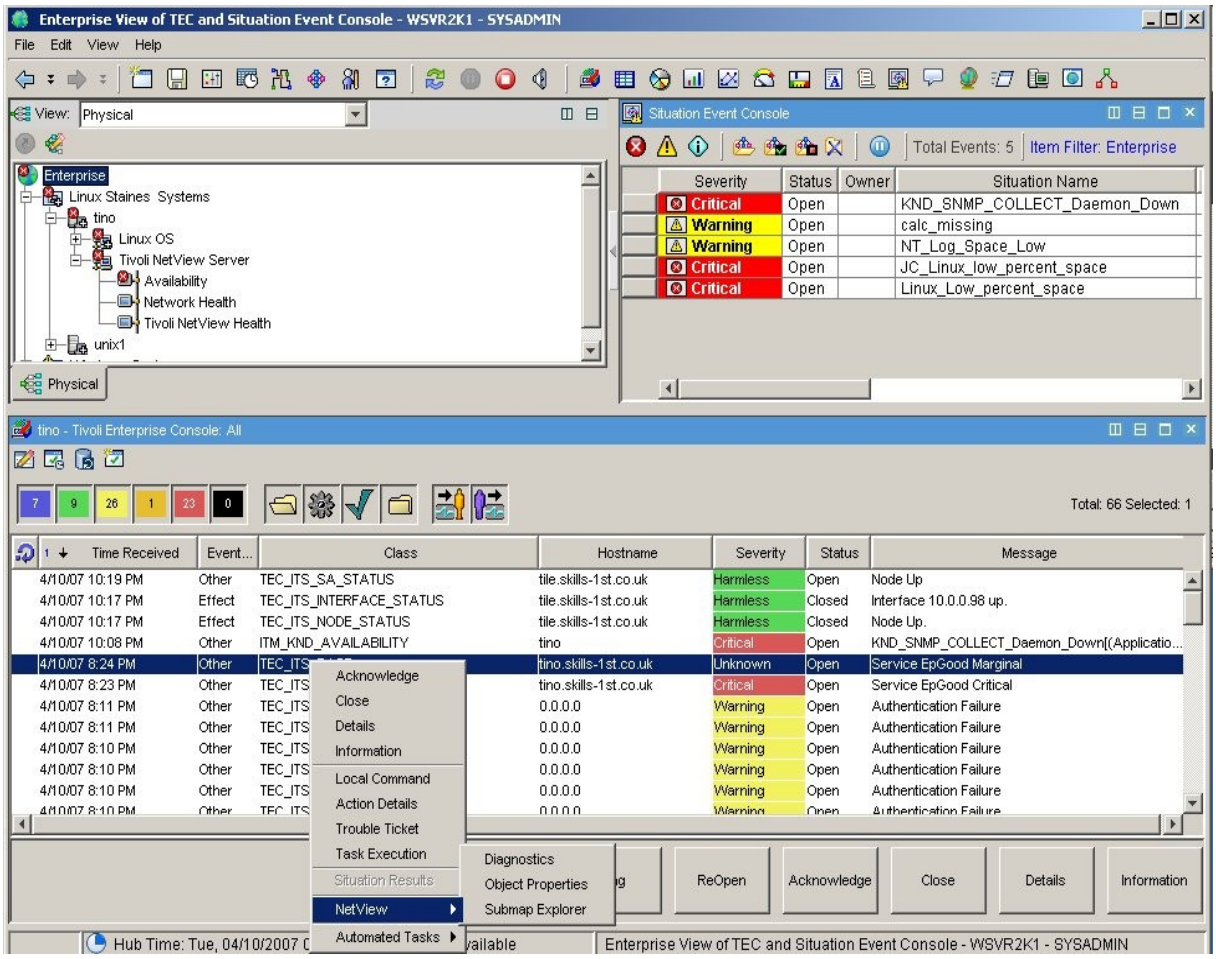

Figure 16: NetView menus from right-click menu of TEC Console in TEP Console

Note that user-configured buttons in the Java TEC Console also work from within a TEP view.

3.3.2 TEC rulebase configuration for the NetView TEMA

When the NetView TEMA Application Support is installed on an ITM 6.1 TEMS, a file, **knd.baroc** is installed; on a Windows TEMS this file will be in $CANDLEHOME\cms\TECLIB; on a Unix TEMS it will be in $CANDLEHOME/tables/<TEMS managed system name>/TECLIB . knd.baroc provides TEC class definitions for all the NetView TEMA situations:

KND_Base ISA Sentry3_5_Base

ITM_KND_AVAILABILITY ISA KND_Base

ITM_KND_PERFORMANCE_OBJECT_STATUS ISA KND_Base

ITM_KND_NODES_UP_DOWN ISA KND_Base

ITM_KND_SMARTSETS ISA KND_Base

ITM_KND_NODES_STATUS ISA KND_Base

```
ITM_KND_NETWORK_OBJECTS   ISA   KND_Base
ITM_KND_TRAP_RATE   ISA   KND_Base
ITM_KND_WEB_CONSOLES  ISA   KND_Base
ITM_KND_NETVIEW_VERSION  ISA   KND_Base
ITM_KND_PING_AND_SNMP  ISA   KND_Base
ITM_KND_TRAP_QUEUE_SIZE  ISA   KND_Base
ITM_KND_DNS_RESPONSE  ISA   KND_Base
```

This file must be moved to the TEC Server and incorporated into the active rulebase. Assuming the file has been copied to /tmp/knd.baroc on the TEC Server, and that the active rulebase is called nvitm, the following commands will activate the NetView TEMA class definitions on the TEC Server. Note that these commands stop and start the TEC so an appropriate time must be selected!

- cd /tmp
- wrb -imprbclass knd.baroc nvitm
- wrb -comprules nvitm
- wrb -loadrb nvitm
- wstopesvr
- wstartesvr

If your TEC is on Unix and your ITM 6.1 TEMS is on Windows, take care with case-sensitivity on the name knd.baroc – you may find it is KND.baroc!

TEC events have six different severities, whereas ITM 6.1 situations only have three. Mapping of event severity between ITM and TEC can happen in two ways:

- A standard procedure with ITM 6.1 situations is that if the ITM Situation name ends in _Cri, _Crit or _Critical the TEC event will automatically have a Critical severity. If the ITM Situation name ends in _Warn or _Warning then the TEC event will automatically have a Warning severity.
- To explicitly map the severity of any situation event to a TEC event, edit the file **tecserver.txt** on the Hub TEMS. By default, this file will be in $CANDLEHOME\cms\TECLIB on a Windows TEMS and in $CANDLEHOME/tables/<TEMS managed system name>/TECLIB on a Unix TEMS. Documentation and a sample tecserver.txt are supplied in Appendix A of the NetView Server Agent User Guide.

Note that there are no entries in any of the NetView agent-related baroc files, that set a default severity; thus if severity is not specified when being sent by

the Hub TEMS, severity will default to Warning, as specified in the TEC base event.

Appendix A of the NetView Server Agent User Guide details the mapping between NetView Situation events and their corresponding TEC events.

A more generic method of mapping any situation event and event attribute to a TEC class and class slot, requires the use of a *mapping* file (see the ITM 6.1 Administrator's Guide, pages 37 – 40). No explicit mapping file is provided for the NetView TEMA. This means that TEC classes are generated according to the **Attribute Group** that the situation is based on, prefaced by **ITM**. Slots in the TEC class take the ITM agent **Attribute Name**. For example a situation event about NetView Trap Queue Size is based on the Attribute Group KND_TRAP_QUEUE_SIZE which has an Attribute called trap. This situation event will translate to a TEC class called ITM_KND_TRAP_QUEUE_SIZE with a slot called trap.

If TEC and ITM 6.1 are integrated, the installation of the ITM 6.1 TEC Synchronization code on the TEC Server will automatically include a ruleset called omegamon.rls. This ruleset ensures that if events are Acknowledged or Closed at the TEP Situation Event Console, then the events in TEC are similarly Acknowledged. If events are Acknowledged or Closed at TEC, the events in the TEP Situation Event Console are synchronised to have the same status. Since this mechanism is generic, it works for NetView TEMA events as well as all others.

3.4 Historical data gathering

ITM 6.1 introduces Tivoli Data Warehouse 2.x; this is the historical data gathering facility for all ITM 6.1 TEMAs.

3.4.1 Configuring Historical Data collection

By default, all historical data gathering is turned off. It can be configured for any TEMA *Attribute Group* (it is not currently possible to configure for individual attributes or for individual nodes). Historical data collection for the NetView TEMA is configured in exactly the same way as any other TEMA, using either the History Configuration icon (the one with the clock), or Ctrl-H.

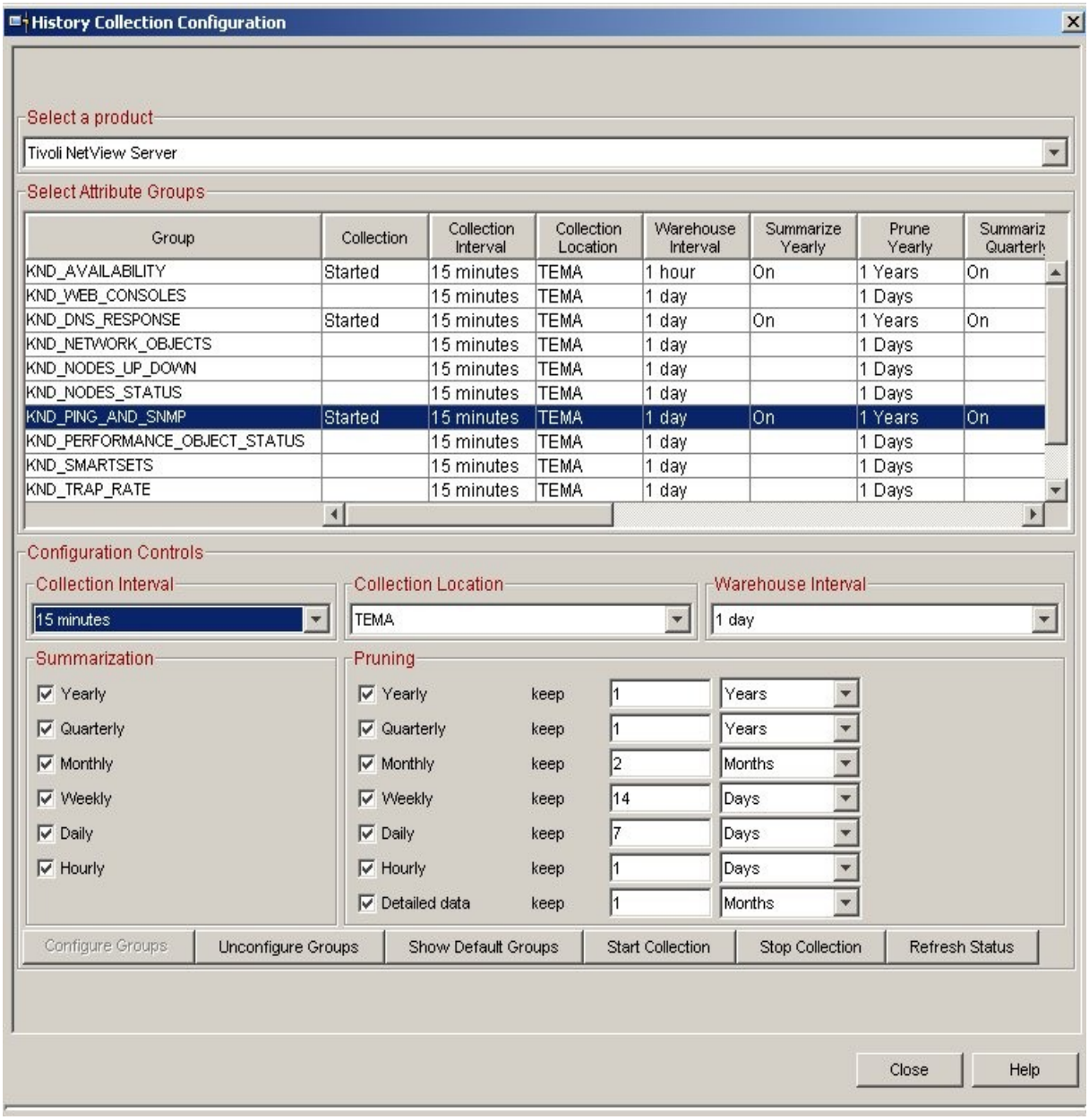

Figure 17: Configuring historical data collection for the NetView TEMA Attribute Groups

The Availability attribute group takes by far the most space in the Data Warehouse database as it has a many-to-many characteristic, ie. many individual attributes are gathered for many processes.

The “Collection Interval” specifies how frequently the NetView TEMA will collect historical data. Short-term data is held at the Unix NetView TEMA in $CANDLEHOME/<architecture>/nd/hist where there will be both a header file and data file for each configured attribute group. An alternative to holding short-term data at the TEMA would be to gather data at the TEMS to which the TEMA is connected; however for many attributes and several TEMAs this can require a large amount of extra diskspace at the TEMS so short-term storage is normally recommended at the TEMA unless firewall

security policies demand otherwise.

The “Warehouse Interval” specifies how frequently the short-term data is pushed via the Warehouse Proxy Agent, into the Data Warehouse database. Possible choices are every hour or every day, thus short-term data should not be older than 24 hours.

The values that are generated by Summarization can be inspected using the native database tools (eg. DB2 Control Centre). Typically, summarisation will generate maximum, minimum, total and average values for most numeric attributes.

If database space is a concern, it would be advisable not to summarise Availability data and to prune Availability detailed data as frequently as is possible, commensurate with local archive policies.

The “ ITM Tivoli NetView Server Agent User's Guide”, Chapter 5 documents all the NetView TEMA attributes and attribute groups, including which can be collected historically (which appears to be all values). There is also a useful section at the end of Chapter 5, page 28, that provides planning information for database sizing for the NetView TEMA attribute groups.

3.4.2 Viewing Historical Data

If Historical Data is available for any configured workspace view, then an icon appears at the top left corner of that view. Clicking the icon provides a panel where you can configure whether to view realtime data or historical data.

If historical data is selected, you can choose whether to display detailed data or summarised data.

Note that there may be several pages of data – the top right corner of the view shows the page(s) available and there are page up/down icons in the right margin to scroll the data.

When constructing views with historical data, it is best to start with a Table view – otherwise you sometimes do not see the top-left, historical data icon. You can always change the view type subsequently.

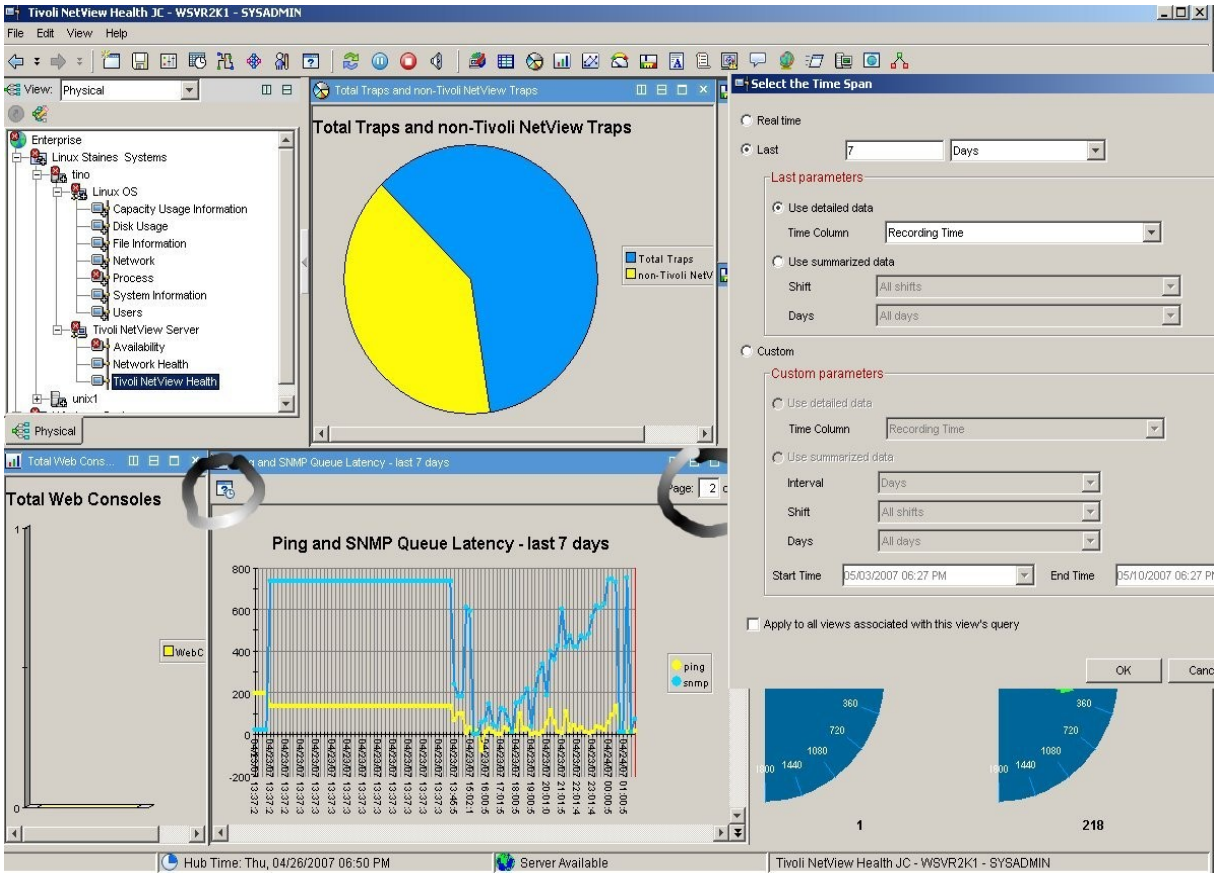

Figure 18: Viewing historical data and the configuration options

3.5 itmquery

itmquery is a standard utility shipped with NetView. Originally designed to provide integration between NetView 7.1.4 and ITM 5.1, it provides a query utility to allow a NetView administrator to view ITM agents and, for ITM 5.1, shows what resource models are distributed to a node. NetView 7.1.5 has extended this functionality to also report on ITM 6.1 TEMAs.

Configuration must be performed on the NetView Server on /usr/OV/conf/itm_servers.conf, either using the **serversetup** utility, or by using the **itmquery --add-server** command. Either way, you need to provide:

- Address for TEPS in the format **http://<TEPS name or address>**
- The TEPS port will default to 1920 – it can be changed to suit local standards
- A valid ITM user / password

This setup can be verified with the **itmquery --verify-server-info** command.

To display the IP addresses of endpoints monitored by the TEPS, use i**tmquery --dump-endpoints** .

Although the verify-server-info command seems to report success, the dump-endpoints command does not appear to work. This is currently the subject of a PMR.

4 What happens under the covers?

The ITM Tivoli NetView Server Agent is shipped as-is. There is a pdf manual that comes with it – " ITM Tivoli NetView Server Agent User's Guide", GC32-1859-00, but this leaves many questions unanswered.

Postings on the NetView "NV-L" mail list make it clear that modifying the functionality of the agent on the NetView side, is not supported and hence the means by which NetView delivers data to ITM is not documented.

This section documents my findings of the underlying workings of the agent, along with some pointers from NetView support and development. Modifying any of this stuff is totally unsupported; similarly, assuming that the functionality will remain the same in future fixpacks, would be dangerous.

4.1 General principles

The ITM NetView agent appears to be built using the Agent Builder technology that is rumoured may ship with ITM 6.2. The Agent Builder provides a wizard-based way of building new data providers for the ITM Universal Agent. It can also generate packages for deploying ITM Workspaces, Situations and Queries that are based on the data gathered by the new data provider.

Typically on a Unix NetView system, the ITM NetView agent code is under $CANDLEHOME/<architecture>/nd/bin; for example on a standard SuSE Linux system, this would be /opt/IBM/ITM/li6243/nd/bin (this directory will be used throughout the examples given here). Scripts on a Windows NetView Server are in C:\IBM\ITM\tmaitm6 , by default.

The "bin" directory contains:

- The agent binary, kndagent
- An XML file with the agent configuration, knd.ref
- A number of shellscript files

The shellscripts in this directory are invoked:

- To populate data into ITM workspace views, either on-demand or periodically

- To deliver values to ITM situations
- To supply data for historical purposes to the ITM Warehouse Proxy Agent

Most of these shellscripts call NetView commands. Many of the NetView commands work by writing data to files in /usr/OV/itm. The shellscripts in /opt/IBM/ITM/li6243/nd/bin typically end by echoing to stdout, the values in the /usr/OV/itm datafiles. These values are received by the kndagent.

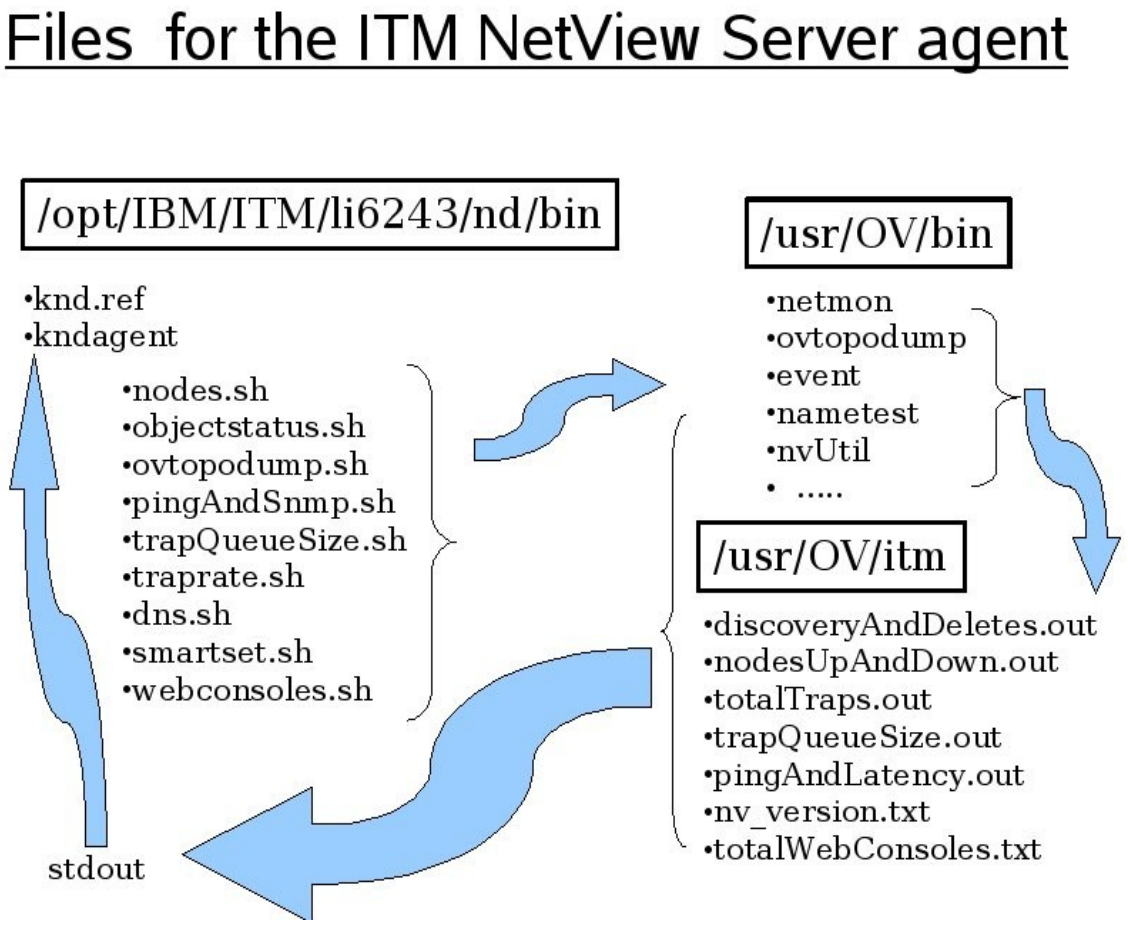

Figure 19: Architecture of NetView TEMA on Unix platform (SLES 9)

Note that many of the shellscripts in /opt/IBM/ITM/li6243/nd/bin send the output from stdout and/or stderr to "null". This results in a file called null in the same directory. Each of these lines should be changed to /dev/null .

4.1.1 New NetView commands to support ITM 6.1

Support for integration between ITM 6.1 and NetView was introduced with NetView 7.1.5 in October 2006. Most of the ITM integration functionality is

accomplished with new, but undocumented, parameters to ovtopodump, and with “netmon -a” actions.

The new parameters are deliberately undocumented and their use by users appears to be unsupported. Please understand that the rest of this subsection documents my experiences and hypotheses – this is **not** supported by IBM.

On Unix NetView platforms, there are now three commands which, I believe, deliver equivalent results. They provide information on NetView internal queues and databases. The numeric parameters for all three commands are the same. The three equivalent methods are shown below – the example of each, dumps the internal pingList- the list that specifies for each interface known to NetView, when that interface will next be polled (in seconds):

- /usr/OV/bin/netmon -a 12
- /usr/OV/service/netmonaction.sh 12 (note – no “-a” in the syntax)
- /usr/OV/bin/event -b openview -e NMAC_EV -a 12

NetView on Windows platforms should support the latter two options (netmonaction.bat is in \usr\OV\bin).

The numeric options are documented in the NetView 7.1.5 Administrator's Reference Guide under the entry for netmon. On Unix platforms, “man netmon” also documents the options well. Note that, although all the NetView documentation has been refreshed for V7.1.5, the new parameters relevant to ITM are **not** documented. (Both Unix and Windows versions of the netmonaction command issued without parameters, outputs a usage message detailing the options that are formally supported).

The scripts used by the ITM NetView agent utilise the third option shown above.

The new parameters for ITM, with my guesses at their functionality, are:

netmon -a parameter	***Function***
745	Writes the number of nodes discovered and the number of nodes deleted, since the last “netmon -a 745”, to /usr/OV/itm/discoveryAndDeletes.out, in the format <Discovered>:<Deleted>
746	Writes to /usr/OV/itm/totalTraps.out, the total number of traps and the number of non-NetView traps, that have occurred since the last “netmon -a 746”. non-NetView traps appear to be defined as **not** coming from the netView6000 enterprise (.1.3.6.1.4.1.2.6.3.1). The format is <Total traps>:<non-NetView traps>

netmon -a parameter	***Function***
747	This parameter appears to generate 2 files. trapQueueSize.out is written to with the single value representing the number of traps waiting to be processed by NetView's trapd daemon. pingAndLatency.out is re-written with 2 values, colon-separated. The first value is the number of seconds before the next interface will be ping'ed by netmon; the second value is the number of seconds before the next node will be SNMP-polled by netmon. Effectively, these values are from the first element on the list generated by a "netmon -a 12", and the first element on the list generated by a "netmon -a 16"

Some agent scripts also use new, undocumented parameters to the /usr/OV/bin/ovtopodump command. In general, ovtopodump is well documented in the NetView 7.1.5 Administrator's Reference Guide, but again, the new parameters are not included.

ovtopodump parameter	***Function***
- I	Writes 8, colon-separated values to stdout. These values represents the number of nodes currently in the NetView topology database, in each of the 8 NetView statuses . The format of the output is <Critical>:<Normal>:<Unknown>:<Marginal>:<Unmanaged>:<User1>:<User2>:<Unreachable>
- U	Writes 2 values to nodesUpAndDown.out. The first value is the number of nodes currently Up (status of Normal). The second value is the number of nodes currently Down (status of Critical). The format is <Up>:<Down>
- T	Writes 5, colon-separated values to stdout. These values represents the numbers of Networks, Segments, Nodes, Interfaces and Gateways currently in the NetView topology database. These values appear to be the same as those reported by the documented "ovtopodump -l" command but in the format <N/ws>:<Segs>:<Nodes>:<I/fs>:<Gateways>

4.2 Workspace views

Four workspaces are delivered, as standard, for any ITM NetView agent. Each of these will be explored.

4.2.1 NetView Server workspace

This is an overview workspace containing five views, most of which are duplicates of views from the other three more detailed workspaces.

The only view that is unique to the NetView Server workspace is the “Tivoli NetView Server Version” table. It simply checks for the presence of /usr/OV/itm/nv_version.txt. If the file exists, it is “cat'ed” to stdout; otherwise an echo command delivers “Tivoli NetView Server 8.1”!

4.2.2 Availability workspace

The ITM Agent Builder toolkit delivers a method to monitor for specific processes and it would appear that this has been used to deliver data for the Availability workspace.

The file knd.ref in /opt/IBM/ITM/li6243/nd/bin documents each of the processes that is to be monitored.

Some of these processes are mutually exclusive; for example NetView 7.1.5 now has two different sets of daemons to collect historical data, snmpCollect and nvcollectord/nvpollerd; snmpCollect and nvcollectord are included in the agent's process list. The User's Guide provides documentation in Chapter 4, page 13, on how to modify the filters of a view to exclude display of processes or to add new NetView processes to be monitored. Note that **only** those processes that are defined to the agent in knd.ref can be monitored – adding a filter in a workspace view for a process of your own, will have no effect.

There are four views:

- Availability (table)
- Processor (bar chart)
- Memory (bar chart)
- Threads (bar chart)

These views are all populated by a single ITM Query called “Availability Data” that brings back all the attributes in the KND_AVAILABILITY attribute group. The “ ITM Tivoli NetView Server Agent User's Guide”, provides a good reference section of all attributes and attribute groups, in Chapter 5.

The Availability workspace does not use any scripts in /opt/IBM/ITM/li6243/nd/bin.

If you wish the NetView TEMA to monitor processes that are not defined in knd.ref it is possible (though probably not supported) to modify knd.ref and

add the required daemon. By default, the new nvpollerd for collecting SNMP data, is not included in knd.ref (although the associated nvcollectord daemon is). The following screenshot shows a line added for nvpollerd. Once knd.ref has been edited, recycle the NetView TEMA. You will need to include a filter in the Availability view that specifies to check for nvpollerd.

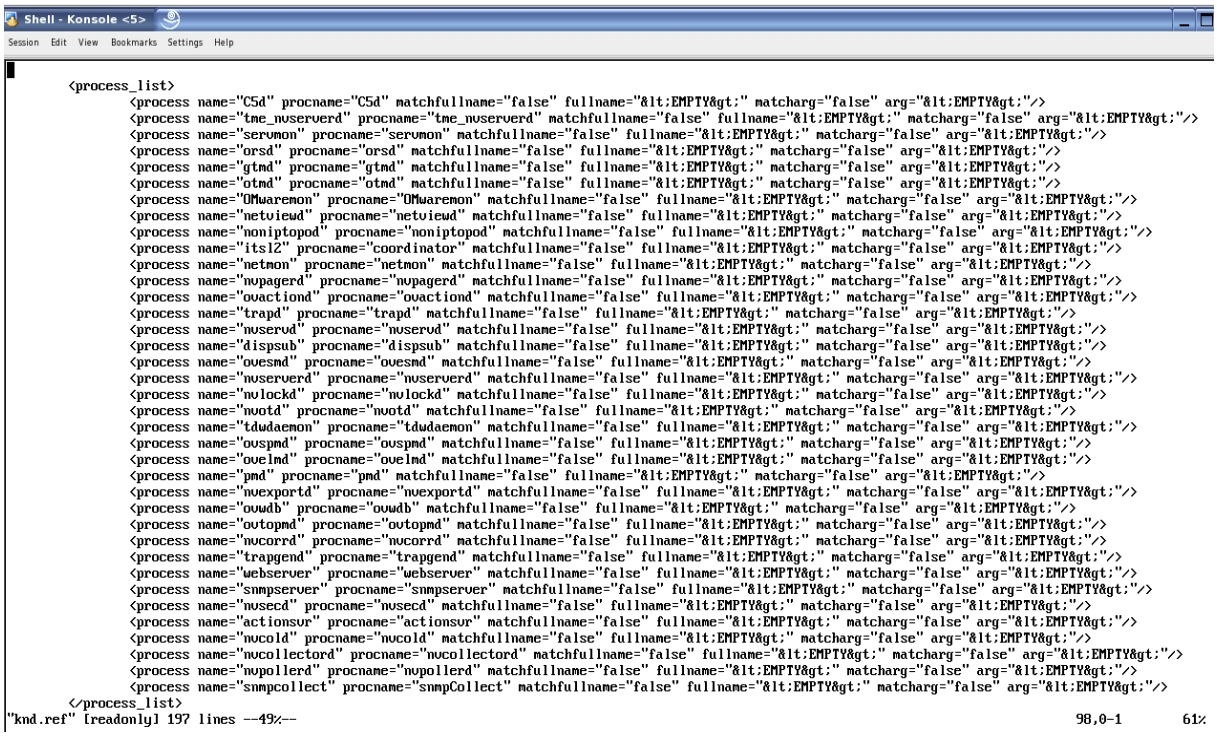

Figure 20: knd.ref showing line included for nvpollerd

Once included in knd.ref, situations can also be written to alert when nvpollerd goes down. (Note that it appears to be impossible to create situation names starting KND_ although this *should* be legal).

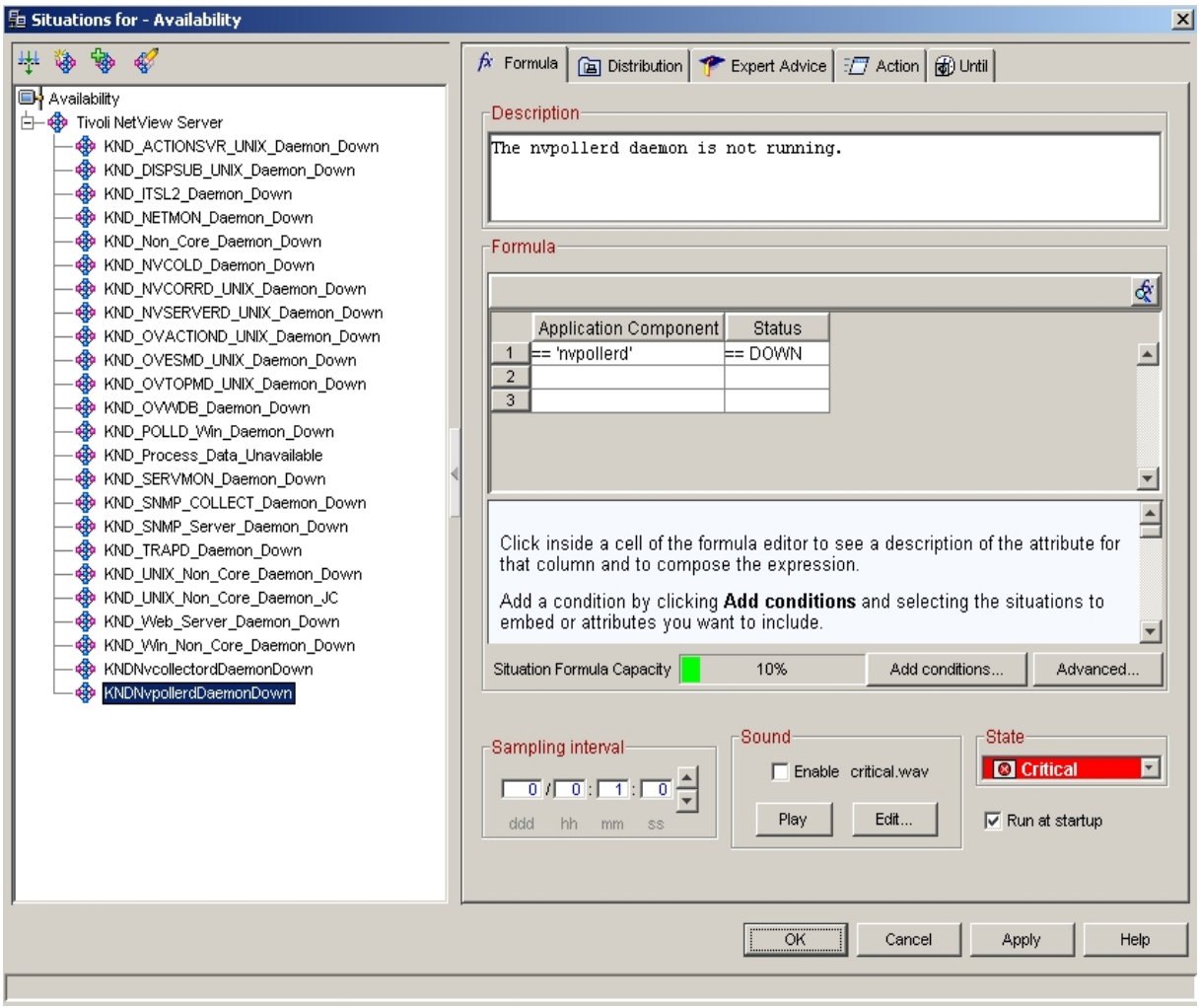

Figure 21: User defined situation to detect nvpollerd down

4.2.3 Network Health workspace

There are four views in the Network Health workspace:

- Node Status
- Nodes Up, Down, Discovered and Deleted
- Network Elements
- SmartSet Membership

Each of these views uses ITM Queries which are instrumented at the NetView system by scripts in /opt/IBM/ITM/li6243/nd/bin. Output is to files in /usr/OV/itm or to stdout.

View Name	*Query*	*Attribute Group*	*Agent script*	*NetView command*	*Output file*
Node Status	Nodes Status	KND_NODES_STATUS	objectstatus.sh	ovtopodump -I	stdout
Nodes Up, Down, Discovered and Deleted	Nodes up down	KND_NODES_UP_DOWN	nodes.sh	event -b openview -e NMAC_EV -a 745 ovtopodump -U	discoveryAndDeletes.out nodesUpAndDown.out
Network Elements	Network Objects	KND_NETWORK_OBJECTS	ovtopodump.sh	ovtopodump -T	stdout
SmartSet Membership	SmartSets	KND_SMARTSETS	smartset.sh	nvUtil (Unix NetView) smartsetutil (Windows NetView)	stdout

4.2.4 Tivoli NetView Health workspace

The Tivoli NetView Health workspace contains five views, as shipped:

- Total Web Consoles
- Total Traps And non-Tivoli NetView Traps
- Trap Queue Size
- DNS Response Time
- Ping and SNMP Latency

Each of these views uses ITM Queries which are instrumented at the NetView system by scripts in /opt/IBM/ITM/li6243/nd/bin. Output is either to files in /usr/OV/itm or to stdout.

totalWebConsoles.txt in /usr/OV/itm is automatically updated whenever a new NetView web console is started. webconsoles.sh simply cat's this file and prints the number of active consoles to stdout. This appears to be the only ITM-related script that has data proactively populated by NetView.

/usr/OV/bin/nametest is a new utility shipped with NetView 7.1.5 to deliver response time of the NetView Server's DNS lookups, in milliseconds (NetView itself has no name-address resolution mechanism – it uses the services of the underlying Operating System). The performance of NetView is very dependent on DNS lookups so this is a very useful utility to help diagnose performance problems with NetView.

nametest is undocumented but appears to take the following parameters:

- -m<number> *the maximum number of lookups to perform*
- -q *quiet*
- -t *test – shows results of each DNS lookup*

Note that there must be no space between the "-m" parameter and the <number>, otherwise the result will always be zero.

/opt/IBM/ITM/li6243/nd/bin/dns.sh uses **-m100 -q** as parameters.

nametest appears to measures the response time of name resolution using the gethostbyname system function, on node names from the NetView object database, chosen at random. The interval for the total lookups is divided by the number of lookups to get a reasonable approximation for DNS response time.

Unfortunately, the base nametest code on Unix has some issues which means that it always delivers the same value! APAR IY97267 addresses two issues with nametest – a temporary fix is available which will be

included with NetView 7.1.5 FP0001. This APAR does *not* yet address the problem that “good” values are returned even if NetView has no access to a DNS resolver!

View Name	*Query*	*Attribute Group*	*Agent script*	*NetView command*	*Output file*
DNS Response Time	DNS response	KND_DNS_RESPONSE	dns.sh	nametest -m100 -q	stdout
Trap Queue Size	Trap queue size	KND_TRAP_QUEUE_SIZE	trapQueueSize.sh	event -b openview -e NMAC_EV -a 747	trapQueueSize.out
Total Traps	Trap Rate	KND_TRAP_RATE	traprate.sh	event -b openview -e NMAC_EV -a 746	totalTraps.out
Ping and SNMP Latency	Ping and SNMP	KND_PING_AND_SNMP	pingAndSnmp.sh	event -b openview -e NMAC_EV -a 747	pingAndLatency.out
Total Web Consoles	Web Consoles	KND_WEB_CONSOLES	webConsoles.sh		stdout

4.2.5 Exporting, modifying and importing workspaces

ITM 6.1 Fixpack 3 introduced some new parameters to the "tacmd" command to list, import and export workspaces. The parameters and their limitations are documented in the Fixpack 3 readme pdf, around page 105. Given the number of views and parameters that the NetView agent typically shows, these new tacmd commands could be very helpful in customisation, though note should be taken of the documented caveats.

Workspaces can be exported / imported using tacmd commands:

- tacmd listWorkspaces | exportWorkspaces | importWorkspaces

Parameters for these commands are:

-s <server>	(your TEPS eg. http://<teps >:1920
-w <workspace name>	(quotes, single or double, protect spaces)
-u <userid>	
-p <password>	
-t <type>	(2 character agent type eg. nd)
-x <filename>.xml	(output xml file)
-f	(on export / import – force – no confirmation)

Note that there are limitations on workspace export/import documented in the ITM 6.1 Fixpack 3 readme (P105), notably:

- custom queries and custom situations are **not** exported / imported from one TEPS to another. (They could be explicitly exported / imported using tacmd viewSit | createSit).
- export / import does not work in a *logical* navigator unless exactly the same navigator items are in the view.

Situations that have been associated with a Navigator item are **not** affected by the export / import of a workspace associated with that item, provided the export / import are to the same TEPS.

Note that the user needs both Workspace Author mode and Workspace Admin mode. For this reason, it may be appropriate to create a new ITM 6.1 user solely and explicitly for performing workspace imports and exports. Any TEP that is running after a workspace import has been performed, seems to need to logoff and log back on again before the new workspace can be seen. If an active TEP user is in Workspace Admin mode, then coming out of Admin mode suffices to provide the newly imported workspace. I have created the user *expimpws* which is cloned from the sysadmin user but, in addition, has the Workspace Admin permission mode set.

The following scenario will logon to the ITM 6.1 TEMS environment, list NetView agent workspaces, export the workspace called “Avail Jane” and then reimport that workspace.

- tacmd login -s wsvr2k1 -u expimpws -t 1440
- tacmd listWorkspaces -u expimpws -t nd
- tacmd exportWorkspaces -u expimpws -t nd -w “Avail Jane” -x avj.xml
- < details of views in the xml file could then be modified>
- tacmd importWorkspaces -u expimpws -x avj.xml
- tacmd logout

If the “-p” parameter is not specified you will be prompted for the user's password (which will **not** be echoed).

Unless the “-f” (force) parameter is used you will be prompted to confirm the actions you want to take place. If you are overwriting an existing workspace, you will be prompted for confirmation again.

Modifying the xml file before reimporting should be done with extreme care and, preferably, with a backup. Corrupting this file may result in the loss of the workspace entirely.

5 Using the ITM TEP to display NetView SNMP data

With NetView 7.1.5 there are now two, mutually exclusive options for gathering SNMP data from agents:

- **snmpCollect** – this is the old system that (at least via GUI menus) can only collect SNMP V1 MIB data. You can collect SNMP V2 MIB data if you edit /usr/OV/conf/snmpCol.conf directly. snmpCollect is strictly limited to using the SNMP V1 protocol. Data is held in the NetView snmpCollect “database” in /usr/OV/databases/snmpCollect – the data is held in Operating System files. It can be externalised using the snmpColDump utility to output to ASCII files (snmpodump for NetView on Windows). The snmpCollect data can be exported to an external database using the RDBMS Interface Module (RIM) functionality of the Tivoli Framework (if the NetView Server is on AIX or Solaris) or it can export to a Tivoli Data Warehouse (TDW) 1.x environment using the tdwdaemon (all NetView Server versions).
- **nvcollectord / nvpollerd** – this is the new system. It can collect both SNMP V1 and SNMP V2 MIB variables, configured via the new **nvcollector** GUI. It can also use either V1 or V2 of the SNMP protocol. Data is stored in a local DB2 database. Data can be externalised into an ASCII file using the snmpdb.sh utility. There is **no** integration with either RIM or TDW 1.x.

5.1 Overview of new nvcollectord / nvpollerd

The nvpollerd daemon is one of two daemons used by the SNMPv1/v2 SNMP Collector application. The SNMP Collector application is a graphical user interface provided with the Tivoli NetView program to handle MIB data collection.

The nvpollerd daemon does the following:

- Retrieves the data specified in the polling configurations from the polled target hosts
- Performs any calculations necessary for the various expressions in the configuration
- Stores the collected data results in the SNMPDATA database for subsequent display by the nvcollector application

The nvpollerd daemon uses the properties specified in the /usr/OV/conf/nvpollerd.properties file to perform these functions. You can change the properties by editing the file. If you make any changes to the properties file, you must restart the nvpollerd daemon for the changes to take effect.

The nvcollectord daemon does the following:

- Receives configuration information from the nvcollector application. Configuration information consists of the targets for which data should be collected, the data to be collected (known as the MIB expressions), and the schedule or rate of data collection.
- Resolves membership in SmartSets. For example, if a Tivoli NetView user specifies that " bandwidth utilization" needs to be collected from the CiscoRouters SmartSet, the nvcollectord daemon communicates with the Tivoli NetView program to assemble the actual list of hosts that make up the CiscoRouters SmartSet.
- Performs wildcard matching of targets
- Retrieves from the Tivoli NetView program the polling parameters to use for various targets as appropriate. Polling parameter values can be overridden if specified by a user. Polling parameters include, for example, the community string to use to access SNMP MIBs, the number of retries to attempt, and the SNMP version to use.
- Checks for configuration changes according to the timer schedule. If a change is detected in either the targets or the polling parameters, the nvcollectord daemon sends a new expanded polling configuration to the nvpollerd daemon.
- Forwards the expanded polling configuration to the nvpollerd daemon for data collection.The nvcollectord daemon uses properties specified in the /usr/OV/conf/nvcollector.properties file to perform these functions. If necessary, you can change the properties by editing the

file. For example, if the /usr/OV/log/nvcollectord.log shows that you are getting too many timeouts, you can increase the socket timeout value for better results. If you make any changes to the properties file, you must restart the nvcollectord daemon for the changes to take effect.

5.2 Displaying NetView SNMP data at ITM TEP

ITM 6.1 has a standard mechanism to allow workspace views to be populated by data from an SQL database. Fundamentally, workspace views display data that is collected by ITM **Queries**. Queries can either be product-provided or created by users. Each ITM TEMA ships with a number of queries based on the attributes that that TEMA can provide; for example, the NT Logical Disk Query for the Windows Operating System TEMA provides a number of data attributes pertinent to Windows disks. The manual for each TEMA devotes a chapter to documenting the product-provided queries.

An alternative to using queries based on standard TEMA attributes is for users to create **Universal Data Provider** queries. This is a way of collecting data from any ODBC-accessible database, using SQL statements. The SQL statements are issued directly from the TEPS to the database; thus data is returned directly to a TEPS Workspace View and the data is **not** available as normal TEMA attributes. This means that you **cannot** build situations based on these SQL queries. It also means that this technique is limited to TEPS that are based on Windows architecture.

5.2.1 Setting up ODBC

To be able to create queries based on SQL, the TEPS must have an ODBC **System** Data Source Name (DSN) setup – this is purely Operating System functionality, nothing to do with ITM, and is generally performed from the "Administrative Tools" menus.

You will need to have the appropriate ODBC driver installed at your TEPS; the "IBM DB2 ODBC DRIVER" was correct for my environment and already existed as DB2 is used on the Windows system to provide the database for TEPS itself and Data Warehouse. The same driver works to connect to the NetView DB2 database residing on a Linux system.

Provide a unique "Data source name" (I used snmpdata) and click the "Add" button to create a "Database alias" (I used snmpdata again); a description can also be supplied. You are then presented with a panel to supply the details for accessing the database.

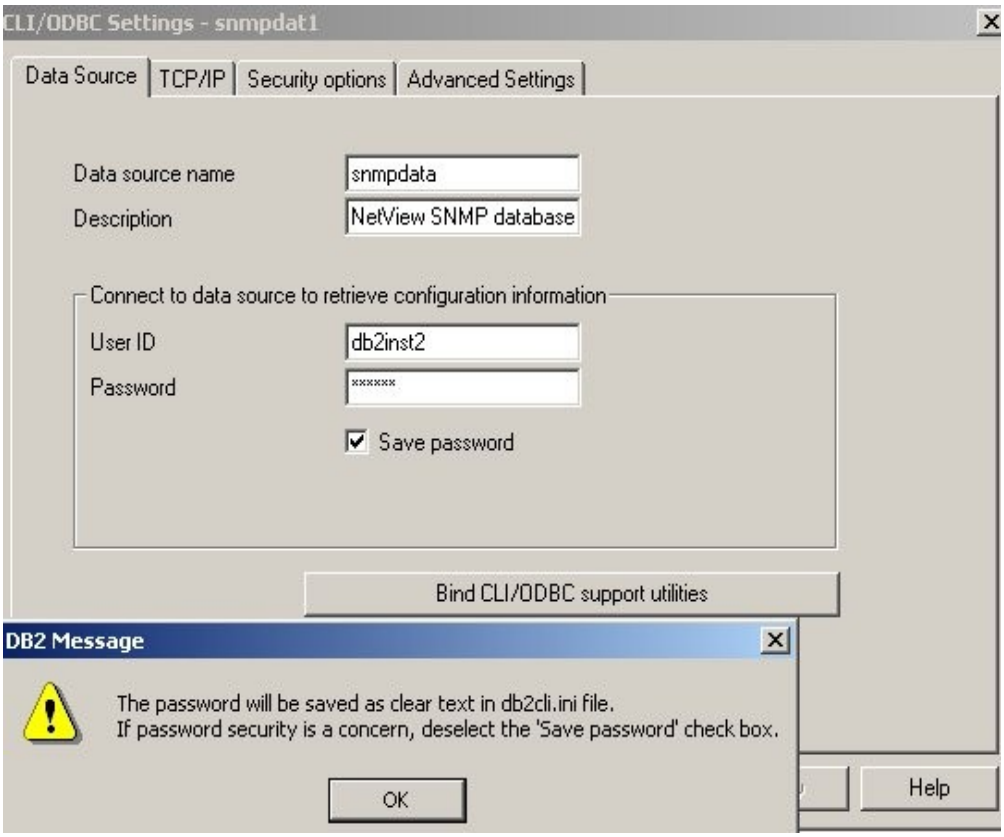

Figure 22: Configuring the snmpdata System DSN

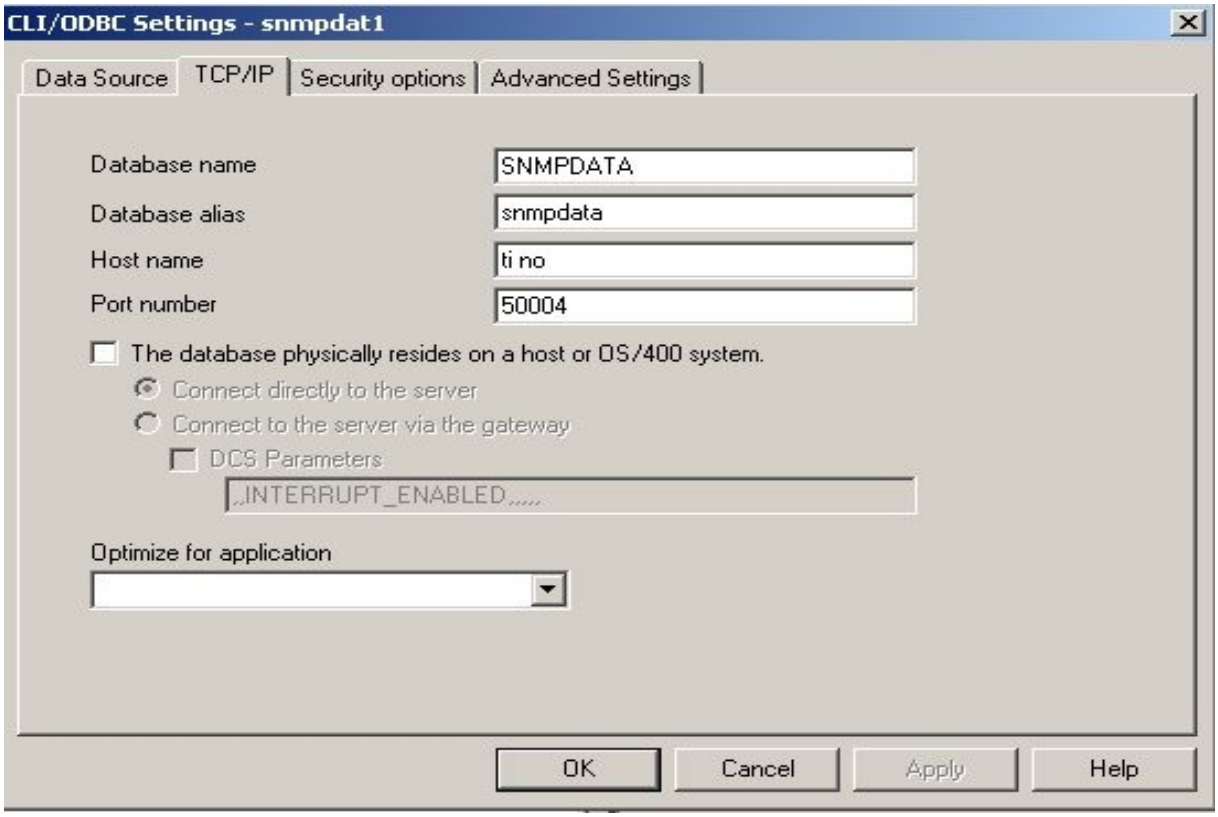

Figure 23: Configuring the remote details for the snmpdata DSN

Provide the userid and password that has full control over the NetView database. You will also need to click the "TCP/IP" tab to specify *where* the database resides.

To find the correct Database name to populate this configuration window, look in **/usr/OV/conf/nvpollerd.properties** on the NetView Server system. This contains the DBName parameter (default is SNMPDATA). The Host name is the name of the NetView Server. The Port number can be found on the NetView Server in **/usr/OV/conf/pollsec.properties**. The default nvcollector.database.port parameter is equal to 50004.

Once the configuration is complete, if you return to the original "ODBC Data Source Administrator" panel, which shows each of your configured System DSNs, you can select the snmpdata DSN, click the "Configure" button, and use the "Connect" button to test that a connection can be successfully made.

5.2.2 Configuring TEPS for ODBC-based Queries

In order for the TEPS to be able to interrogate databases using SQL, the TEPS environment file must be edited and the TEPS recycled. This can be done through the graphical "Manage Tivoli Monitoring Services" application

(right-click the TEPS -> Advanced -> Edit ENV File menus). Add a line to the bottom of this file with:

- DSUSER1="DSN=SNMPDATA; UID=db2inst2;PWD=ibmdb2"

where the UID and PWD parameters match your ODBC configuration, which in turn, match your NetView database user and password. You will need to stop and start the TEPS before this configuration takes effect. Note that if you already have a DSUSER1 line configured, you can have upto 9 such lines, DSUSER1 though DSUSER9 .

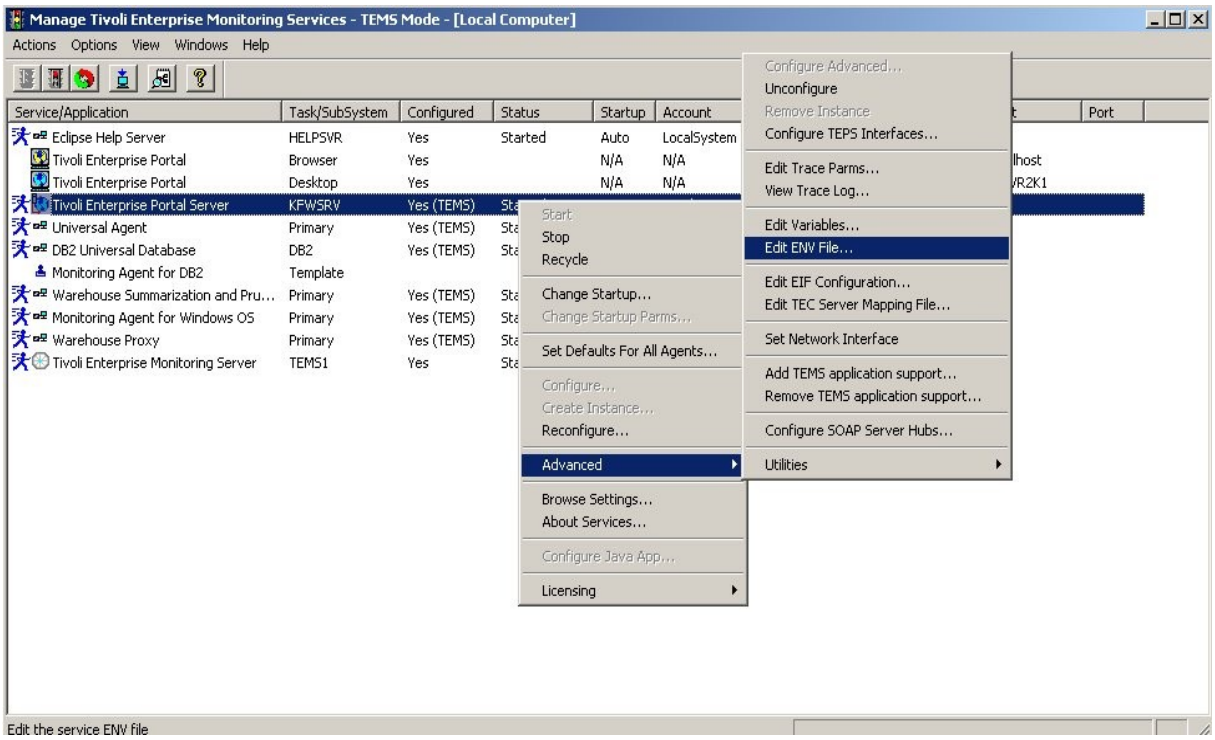

Figure 24: Configuring the DSUSER1 parameter in the TEPS environment file

5.2.3 Creating Queries based on ODBC databases

To create Queries based on ODBC database SQL commands, start the Query Editor in the TEP. You can use the Query icon (the one with the ? symbol), Ctrl-Q or the Edit -> Queries menu.

To create a new Query, explode the "Universal Data Provider" -> Custom_SQL " subtree. You can create a new query with the Create Query icon (the first icon at the top). If you already have a query similar to what you want, the second icon copies an existing query. Provide a unique Query name , select the "Universal Data Provider" category and indicate the correct Data Source.

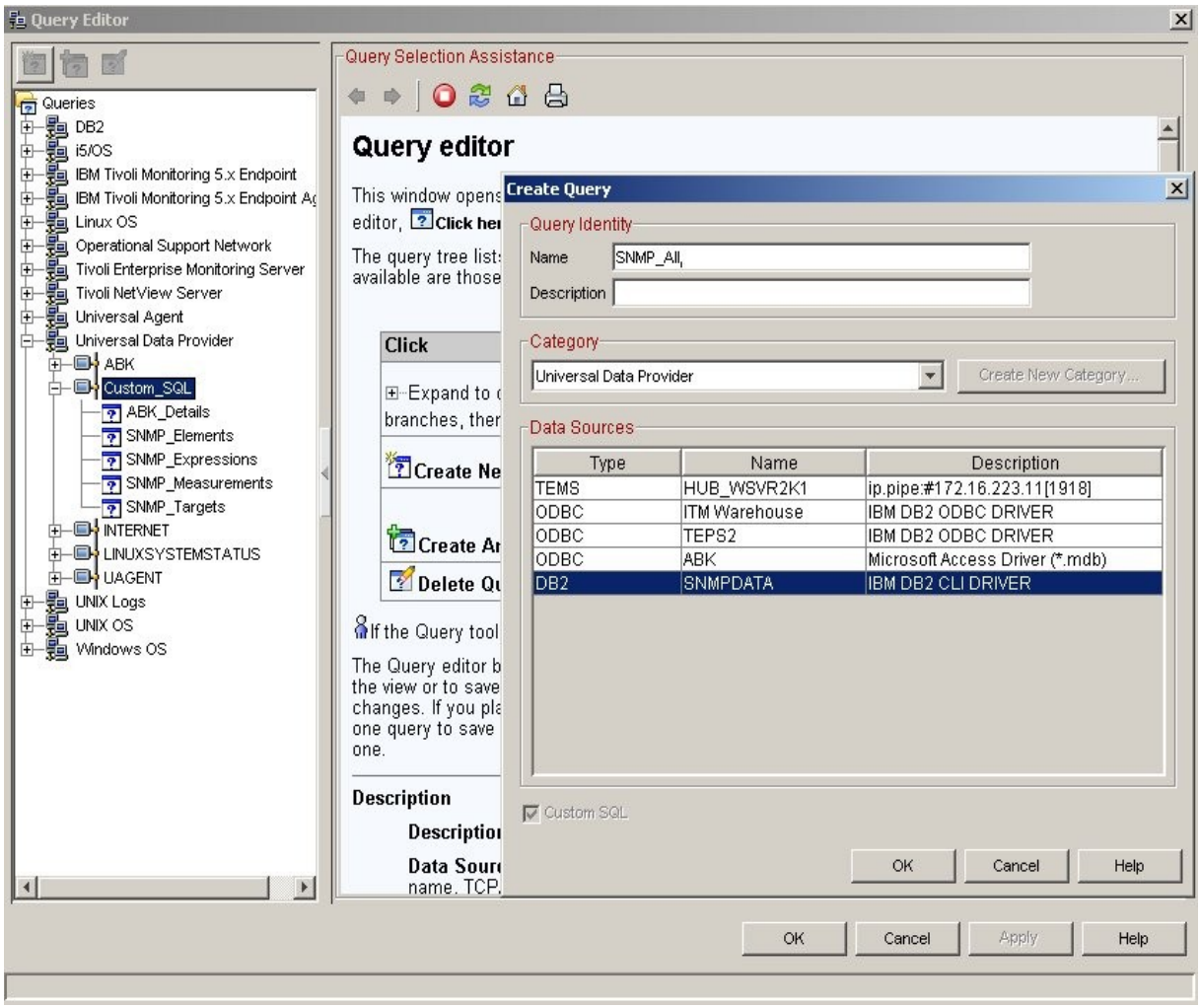

Figure 25: Creating Custom_SQL Queries

You will then need to supply the appropriate SQL statement in the Custom SQL panel.

NB if these queries disappear, create them under DB2-> Custom_SQL, rather than Universal Data Provider-> Custom_SQL.

5.2.4 Creating workspace views to show NetView SNMP data

To display NetView SNMP data in a TEP workspace view, uses exactly the same methods as any other view. Each view in a workspace has data provided by a query. I have created simple queries for each of the four tables in the SNMPDATA database:

- select * from snmpcollect.expression
- select * from snmpcollect.target

- select * from snmpcollect.element
- select * from snmpcollect.measurement

A workspace can then be created with four tabular views, each powered by one of the queries.

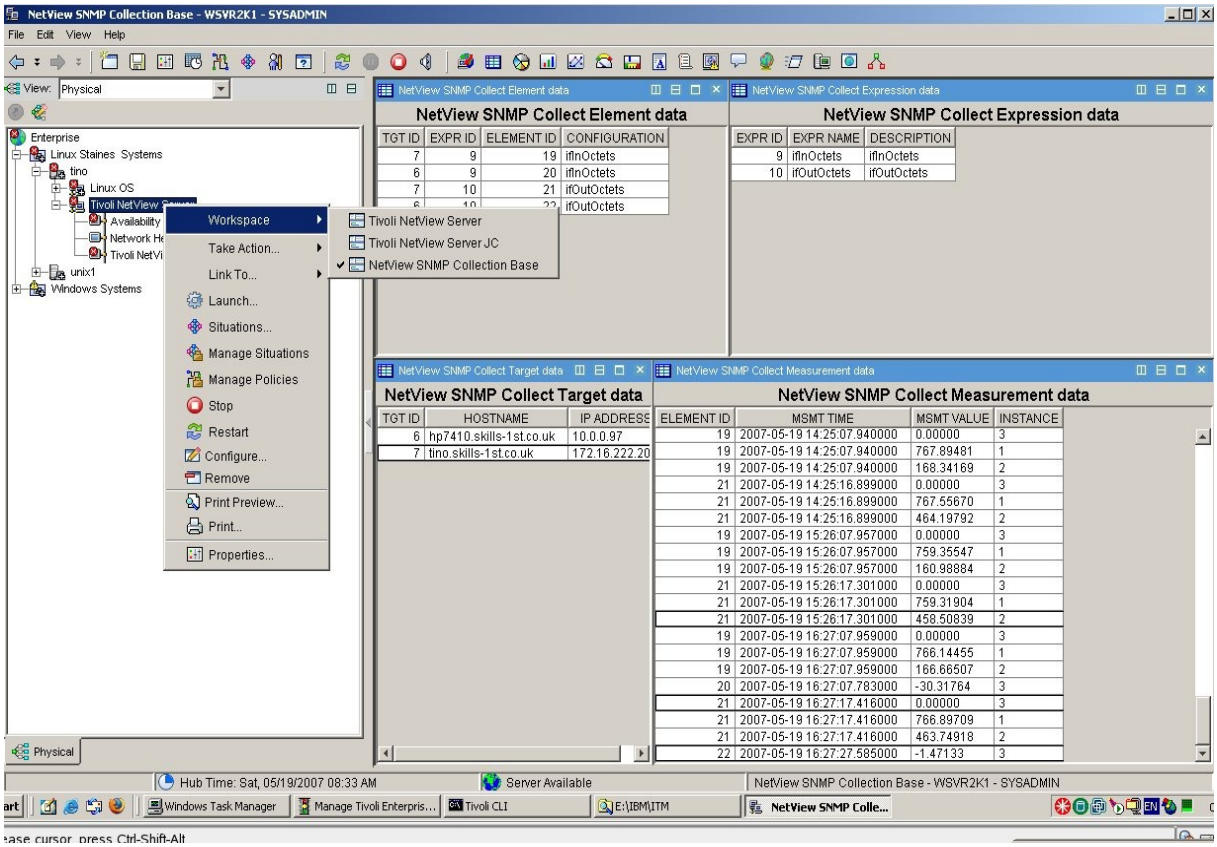

Figure 26: NetView SNMP Collection Base workspace with 4 views

The workspace has been created from the tino Tivoli NetView Server navigator item. You may have any number of workspaces associated with a navigator item; to denote the default, bring up the Properties page for the workspace (from the Edit -> Properties menu).

Appendix A – Useful ITM 6.1 commands

The ITM 6.1 Installation and Setup Guide, GC32-9407, Appendix E, gives a complete command reference.

Here are some of the more common tacmd commands pertinent to manipulating the NetView TEMA.

tacmd login -s <TEMS> -u <user> -t 1440 *logs into TEMS to perform commands*

tacmd listSystems *shows all TEMAs known to this hub TEMS*

tacmd viewDepot *displays packages available at the TEMS for remote install*

tacmd listBundles -i <CD>/ITM/unix *shows bundles available on a CD directory*

tacmd removeBundles -i <CD>/ITM/unix -t nd *removes packages from depot*

tacmd stopAgent | startAgent | restartAgent -t nd *stop or start a NetView TEMA*

Here are some of the more common itmcmd commands pertinent to manipulating the NetView TEMA. Note that itmcmd is only present on **Unix** ITM 6.1 systems and is found in $CANDLEHOME/bin .

itmcmd manage *start "Manage Tivoli Monitoring Services" GUI*

itmcmd agent stop | start nd *stop or start a NetView TEMA*

itmcmd support -t <TEMS> nd *configure Application Support on TEMS*

itmcmd server stop | start <TEMS> *stop or start the TEMS*

References

IBM ITM 6.1 manuals can be found at
http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/index.jsp?toc=/com.ibm.itm.doc/toc.xml

ITM 6.1 Installation and Setup Guide, GC32-9407

ITM 6.1 Administrator's Guide, GC32-9408

ITM 6.1 User's Guide, GC32-9409

ITM 6.1 Fixpack 003 Readme and Documentation Addendum

IBM NetView manuals can be found online at
http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?toc=/com.ibm.itnetview.doc/toc.xml

NetView 7.1.5 Administrator's Guide, GC32-1837

NetView 7.1.5 Administrator's Reference, GC32-1838

ITM Tivoli NetView Server Agent User's Guide, GC32-1859

Good online sources of ITM 6.1 and NetView information are:

nv-l mail list archive

http://lists.skills-1st.co.uk/mharc/html/nv-l/

tme10 mail list archive

http://tme10.uio.no/Apps/Tivoli-List.nsf/main?OpenView

Tivoli User Group website

http://www.tivoli-ug.org/

especially the Tivoli Information Exchange which has ITM and NetView sections

NetView Tivoli User Group website

http://www.nv-l.org/

Gulf Breeze have some excellent material, especially for ITM – check out their BLOG!

http://gulfsoft.com/

Similarly, Orb Data in the UK

http://www.orb-data.com/

Similarly, Capital Software

http://www.capitalsoftware.com/