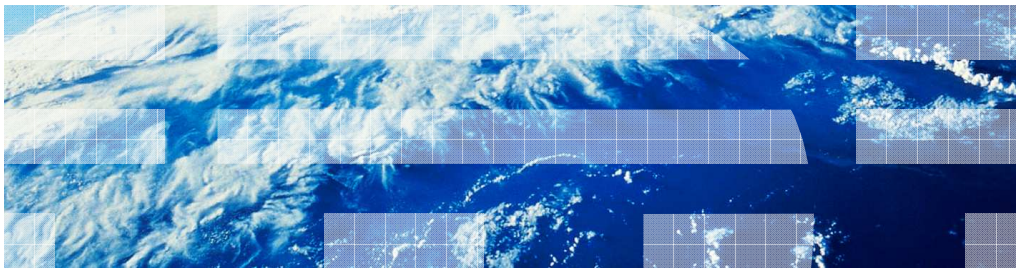
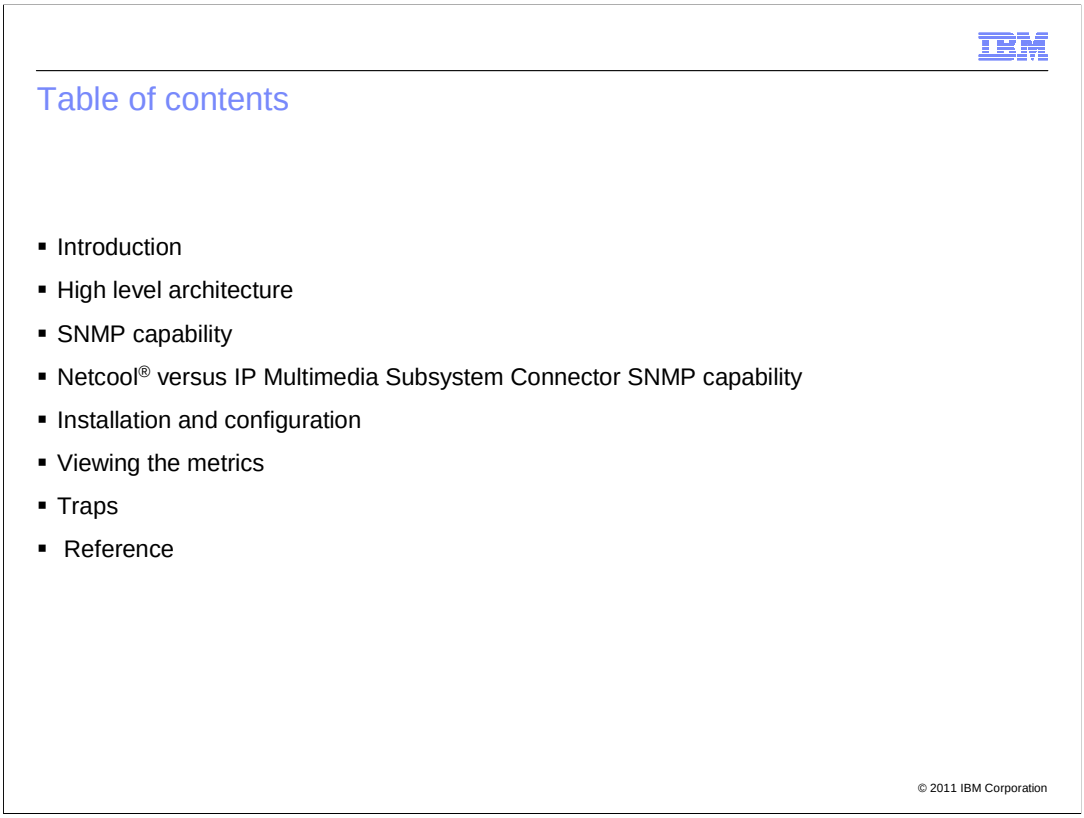
IBM Communication Service Enablers V7.2

IP Multimedia Subsystem Connector SNMP capability

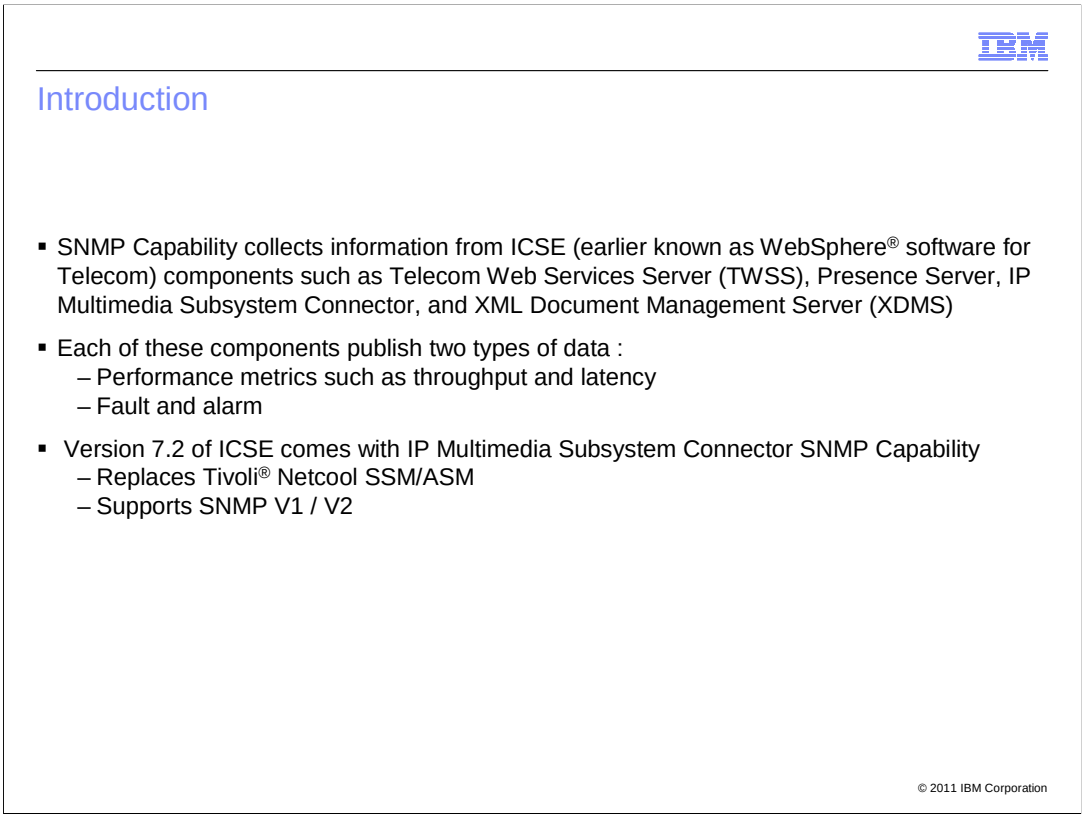


This presentation deals with IBM IP Multimedia Subsystem Connector SNMP (Simple Network Management Protocol) Capability provided with ICSE version 7.2.



The agenda includes:

- Introduction to the SNMP capability
- High-level architecture of IP Multimedia Subsystem Connector
- Features of the SNMP capability
- Comparison between Netcool and IP Multimedia Subsystem Connector SNMP capability
- Installation and configuration of the SNMP capability
- How to view the metrics and understand it
- An introduction to traps
- Reference



SNMP Capability basically monitors the performance of devices over a network.

Previously, ICSE (also known as WebSphere software for Telecom) used Tivoli Netcool. IP Multimedia Subsystem Connector SNMP Capability is a replacement and an enhancement to the features offered by Netcool.

SNMP Capability collects information from ICSE components such as Telecom Web Services Server (TWSS), Presence Server, IP Multimedia Subsystem Connector, and XML Document Management Server (XDMS).

There are two types of data that each of these components publish:

- Performance metrics
- Fault and Alarm

In this release, IP Multimedia Subsystem Connector supports SNMP version 1 and version 2. Version 2 is also called version 2c, where “c” is for the community-based security that version 2 offers.

SNMP overview

- Simple Network Management Protocol
- UDP-based application layer protocol
- SNMP uses an extensible design, where the available information is defined by management information bases (MIBs)
- MIBs describe the structure of the management data of a device subsystem; they use a hierarchical namespace containing object identifiers (OID)
- Each OID identifies a variable that can be read or set through SNMP
- SNMP request and response are in the form of protocol data units (PDUs)
- Each PDU has these fields:
  - IP-header UDP-header version community PDU-type
  - request-id error-status error-index variable-bindings

This slide provides an overview of SNMP.

SNMP stands for Simple Network Management Protocol. It is an UDP-based application layer protocol.

SNMP uses an extensible design, where the available information is defined by management information bases (MIBs).

MIBs can be added and removed as per the requirement. But with this present offering, the WebSphere MIBs essential for SNMP Capability to function properly are packaged.

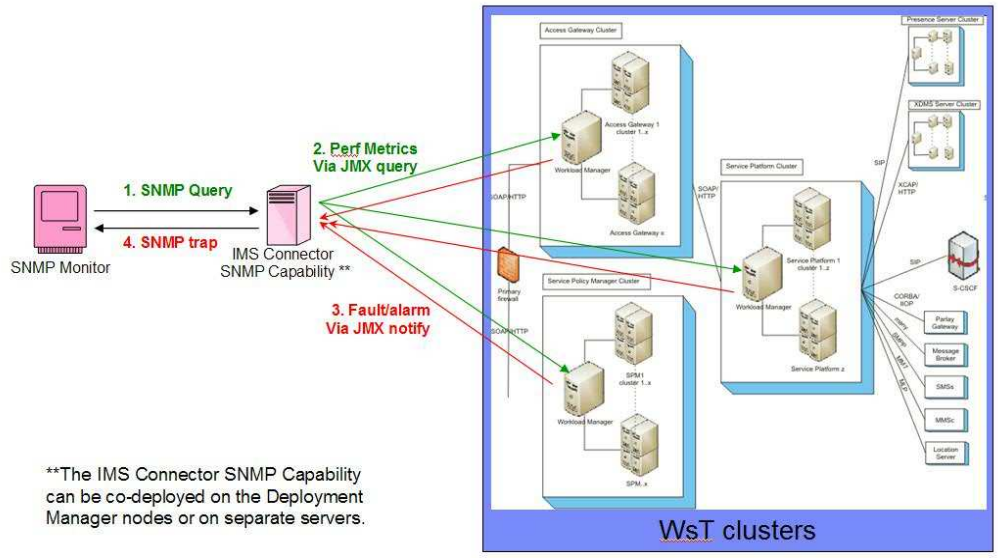
MIBs describe the structure of the management data of a device subsystem; and use a hierarchical namespace containing object identifiers, known as OIDs.

Each OID identifies a variable that can be read or set through SNMP.

This slide also contains the typical protocol data unit or PDU format for SNMP request and response, which contains the IP-header, UDP-header, version community, and so on.

Variable-bindings is the column where the OID and its value is contained.

High-level architecture of SNMP Capability



High-level architecture of IP Multimedia Subsystem Connector SNMP Capability is shown in the diagram.

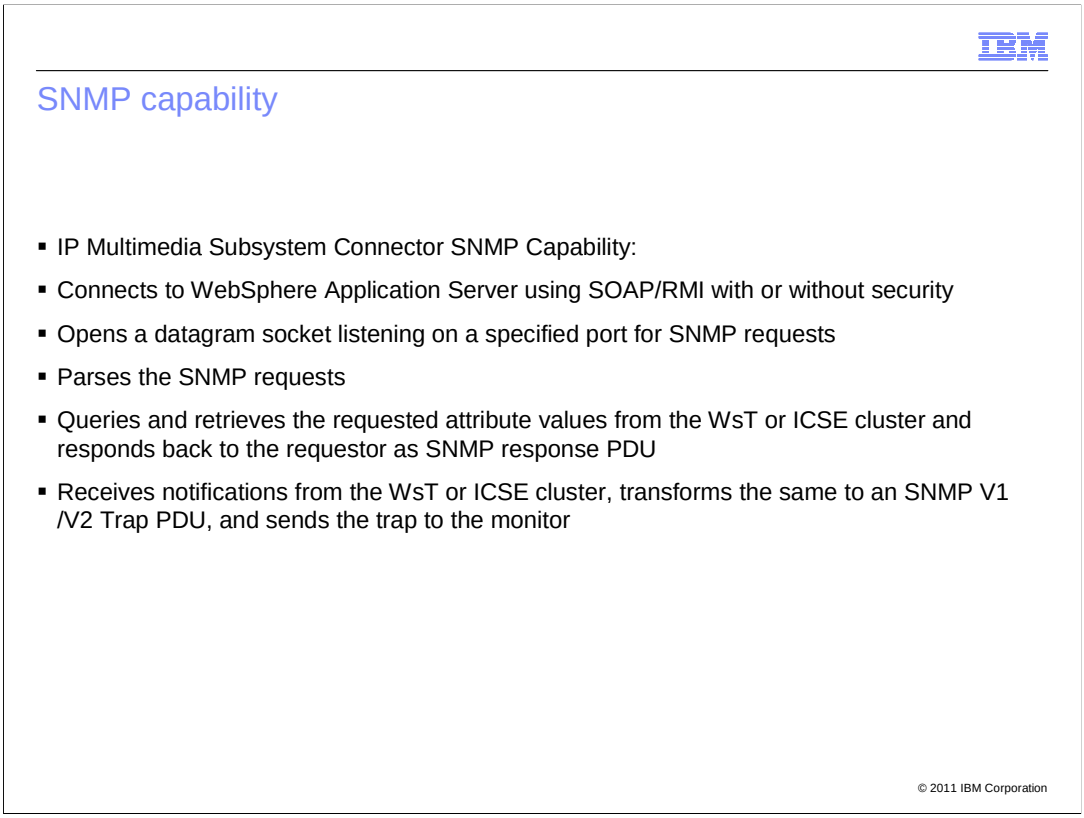
Generally, there is an SNMP monitor provided by the service provider. This presentation takes the example of the MIB explorer that comes along with Tivoli Netcool.

The monitor sends an SNMP query to the SNMP capability, which is the JAR file.

The SNMP capability in turn contacts either the WebSphere Application Server or the WebSphere Enterprise Service Bus and it retrieves data from the PMI infrastructure of WebSphere.

In addition to the Performance metrics, there is the Fault and Alarm feature.

Whenever there is a fault in any of the ICSE (also knows as WsT – WebSphere software for Telecom) components, it sends a JMX notification to the SNMP capability, which in turn converts it into a trap and sends it to the SNMP monitor.



Some of the SNMP Capability features are:

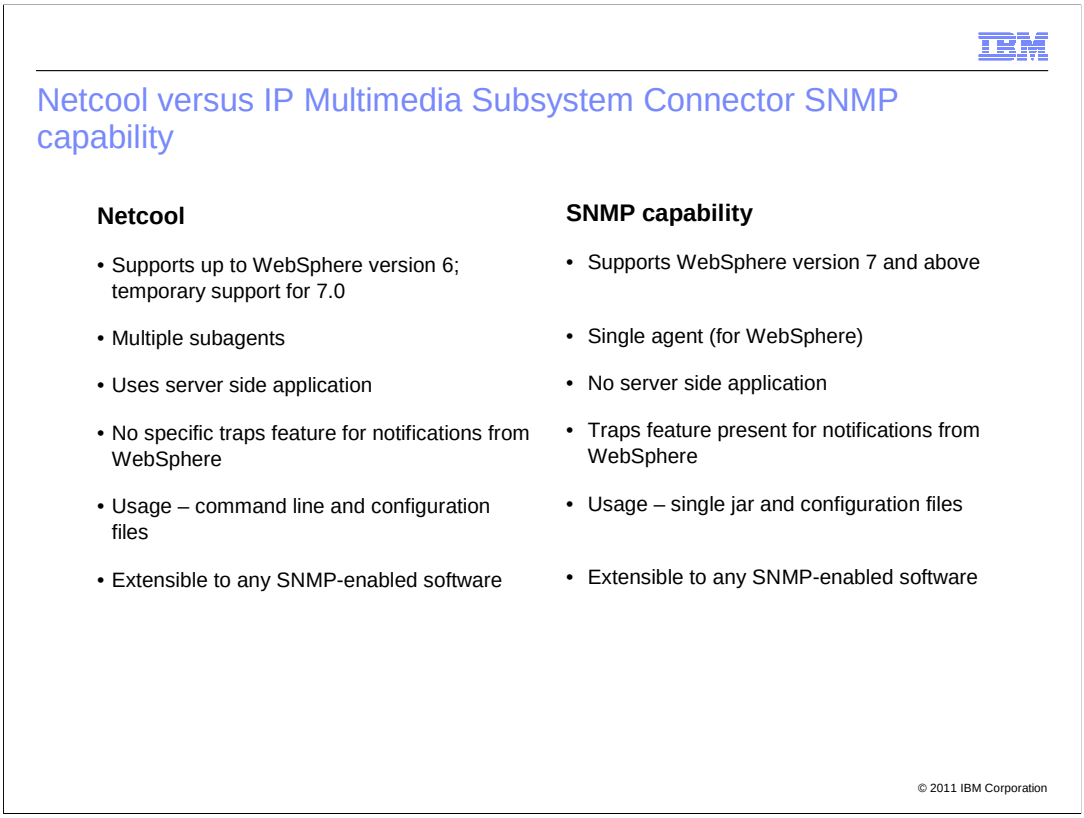
- It connects to WebSphere Application Server using SOAP/RMI with or without security.

- It opens a datagram socket listening on a specified port for SNMP requests.

- When it receives the request, it parses them and looks for the OIDs requested.

- Then it queries and retrieves the requested attribute values from the WsT cluster illustrated in the previous slide.

- It also receives notifications from the cluster, transforms the same to an SNMP v1 /v2 Trap PDU, and sends the trap to the monitor.



Here is a comparison between Tivoli Netcool and IP Multimedia Subsystem Connector SNMP Capability.

Netcool supports up to WebSphere version 6, and it also has a temporary support for 7.0. But going forward, ICSE monitoring support is provided by SNMP Capability.

Netcool has multiple agents for different kinds of application servers, and IP Multimedia Subsystem Connector SNMP capability is only for WebSphere.

Netcool uses a server side application to gather perform metrics, and SNMP capability does not use any server side application.

There is no specific traps feature for notification from WebSphere in Netcool. This is a new feature added in SNMP capability for receiving fault and alarm from ICSE components and transforming them into traps.

The usage is quite similar, except that SNMP capability is a single jar file and configuration files.

Also, both Netcool and SNMP capability are extensible to any SNMP-enabled software.

Installation and configuration

- Pre-installation tasks – JRE version 1.6 and above, WebSphere version 7.0 and above
- Installation – A .zip file containing the SNMP capability jar file and configuration folders
  – (Recommended to install on Deployment Manager machine)
- Configuration – Three XML files agentConfig.xml, jmxConfig.xml and trapConfig.xml
  – agentConfig.xml – contains SNMP capability details
  – jmxConfig.xml – contains details of WebSphere Application Server to be monitored
  – trapConfig.xml – contains trap destination details
- Running the SNMP capability
  – java [-options] –jar SNMPCapability.jar
- Troubleshooting – requires an additional argument while running the jar file
  – java -Djava.util.logging.config.file=config/Logging.properties –jar SNMPCapability.jar

Before you install, check if the JRE version is 1.6 or above, and if the WebSphere version is 7.0 or above for both WebSphere Application Server and WebSphere Enterprise Service Bus.

There is an interactive installer that comes along with IP Multimedia Subsystem Connector that helps you install the SNMP capability.

The installation is straightforward. There is a .zip file that needs to be unpacked to any location on the system. This is done by the interactive installer. The procedure is described in detail in this slide.

It is best that the SNMP capability is installed on the Deployment Manager machine.

The configuration is mainly done through three XML files:

- agentConfig.xml contains the SNMP capability details, such as the host name and the port on which the Java™ process is going to run.

- jmxConfig.xml contains the details of WebSphere Application Server that has to be monitored

- trapConfig.xml contains the trap destination details

Details of the jmxConfig.xml file are mentioned in the next slide.

To run the SNMP capability, run the jar file. Logs can be collected by providing an additional argument as given in this slide.

jmxConfig.xml

```
<jmxConfig
connectorType="SOAP"
security="no"
serverTypes="SOAPmbeanServer RMImbeanServer"
    <SOAPmbeanServer
    address="localhost"
    port="8881"
    sampleInterval = "60"
    user=""
    password=""
    trustStore="/opt/IBM/Websphere/AppServer/profiles/Dmgr01/etc/DummyClientTrustFile.jks"
    tsPassword="WebAS"
    keyStore="/opt/IBM/Websphere/AppServer/profiles/Dmgr01/etc/DummyClientKeyFile.jks"
    ksPassword="WebAS"
    connectorSOAPConfig="/opt/IBM/Websphere/AppServer/profiles/Dmgr01/properties/soap.client.props">
    </SOAPmbeanServer>
    <RMImbeanServer
    address="localhost"
    port="2810"
    user=""
    password=""
    sampleInterval = "60"
    connectorRMIConfig="file:/opt/IBM/Websphere/AppServer/profiles/Dmgr01/properties/sas.client.props"
    sslRMIConfig="file:/opt/IBM/Websphere/AppServer/profiles/Dmgr01/properties/ssl.client.props">
    </RMImbeanServer>
    <mbean>
    <collection counterMode="automatic" counters="" expression=".*"/>
    </mbean>
</jmxConfig>
```


The jmxConfig.xml file contains all the details of the WebSphere Application Server that needs to be monitored.

connectorType - can be SOAP or RMI. It is taken as SOAP by default.

security - is the WebSphere security. It can be "yes" or "no".

If "yes", the user name and password used to log onto the administration console is required.

For the SOAP MBean Server:

- the address is the WebSphere Application Server address

- the port is the SOAP port

- sample interval is recommended to be at least 60 so that performance is not hindered

- the user name and password is required if security is enabled and it is the credentials used to log on to the administration console

There are certain attributes that need to be specified for SSL security, like the path and password to the trust Store and key Store, and the path to the connectorSOAPconfig, which is the properties file provided by WebSphere Application Server for connection using SOAP.

The attributes are similar for RMI.

There is another node called MBean. This is where you can specify the collections or MBeans that have to be monitored.

A collection node will have attributes such as expression, which gives the regular expression that is to be used to map MBeans that are to be monitored.

There is counter mode, which can be automatic or manual.

If it is automatic, all counters of the matched MBean are collected. If it is manual, the counters have to specified in a comma-separated format.

This is the only part of the XML that is dynamic, which means it can be changed even while the capability is running. The other part of the XML can be changed after stopping and starting the capability.

agentConfig.xml

```
<agentConfig name="WsTAGENT" logFile="log/WsTSNMP.log"
   registryPort = "1135" agents="WsTSNMP">
 <metadata mibsFolder = "mibs"  mibs="nh-smi.smi,rfc1213-mib.mib,rmon-mib.mib,snmpv2-
  tc.mib,tokenring-rmon-mib.mib,websphere-mib.mib,rmon2-mib.mib"    />
 <WsTSNMP ipAddress = "localhost" snmpPort = "10162" views="V1"
 Behaviors="getScalars,getCollection,getControl,getCounter,getData,getTrapDestinations">
          <V1   communityNames = "public"
                jmacfgFiles = "oid.out"
                trapConfig = "config/trapConfig.xml"/>
          <getScalars oid = "1.3.6.1.4.1.1977.22.10.1.0"
                    class = "com.ibm.wst.snmp.behaviors.GetScalarsBehavior"
                    properties = "configFile=config/jmxConfig.xml“ />
          <getCollection oid = "1.3.6.1.4.1.1977.22.10.10.1.2"
                    class = "com.ibm.wst.snmp.behaviors.GetCollectionBehavior"
                    properties = "configFile=config/jmxConfig.xml“ />
                  <.....truncated..>
           </WsTSNMP>
</agentConfig>
```

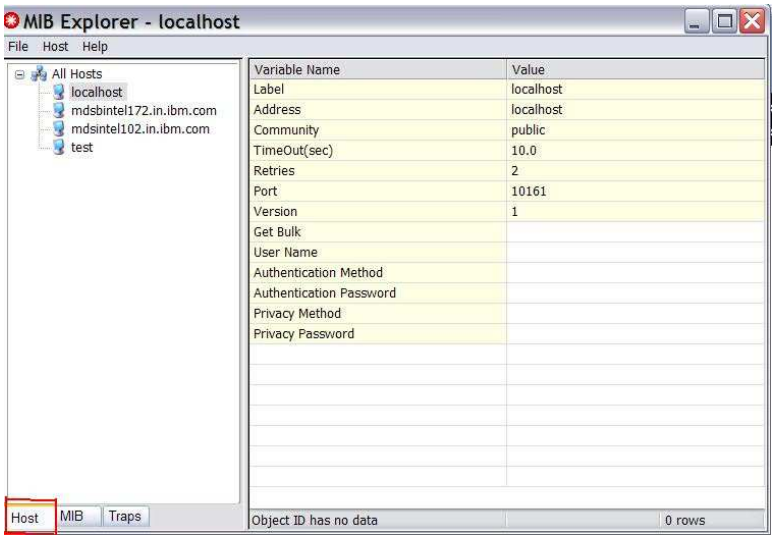

The agentConfig.xml file contains the details about the host where SNMP capability is installed.

The parameters that need to be configured are:

- ipAddress : Fully qualified host name where SNMP Capability is running.

- snmpPort : The port where SNMP capability is listening.

Viewing the metrics (1 of 4)



Once the regular expression is specified in the configuration file, you can view the metrics in any SNMP-enabled software.

This presentation takes the example of the MIB explorer that comes along with Tivoli Netcool.

In this screen capture, you can see three tabs at the bottom - **Host**, **MIB**, and **traps**.

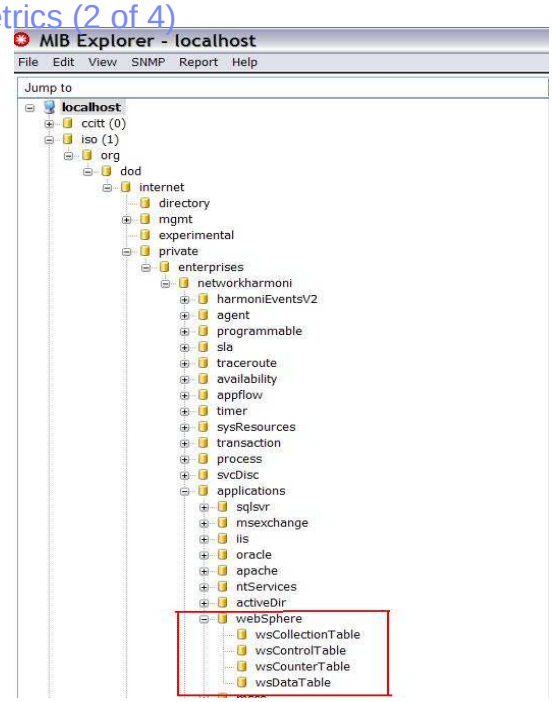
Host has been highlighted in red and its details are shown here.

Any number of hosts can be created here. These contain the details of the machine on which the SNMP capability is running.

The port mentioned is the one that is configured in the agentconfig.xml file.

The request is send to the IP address and port number specified in the **Host** tab.

Viewing the metrics (2 of 4)



Each MIB describes the hierarchical structure and this screen capture shows the table structure for WebSphere.

There are four tables under WebSphere - **collection**, **control**, **counter**, and **data table**.

Viewing the metrics (3 of 4)

Collection table

Index	Name
24	rharish1Node01,server1,hamanagerModule
25	rharish1Node01,server1,hamanagerModule>HAManagerMBean
26	rharish1Node01,server1,jvmRuntimeModule
27	rharish1Node01,server1,objectPoolModule
28	rharish1Node01,server1,objectPoolModule>ObjectPool_ibm.system.objectpool_com.ibm.ws.webcontainer.srt.SRTConnectionContextImpl
29	rharish1Node01,server1,objectPoolModule>ObjectPool_ibm.system.objectpool_com.ibm.ws.webcontainer.srt.SRTConnectionContextImpl.class@11801180
30	rharish1Node01,server1,orbPerfModule

Control table

Index	Status	Owner	Collection	CounterMode
1	active	WsT SNMP AGENT	objectPoolModule.*	automatic

Counter table

WsControlIndex	Index	Name
1	1	ObjectsCreatedCount
1	2	ObjectsAllocatedCount
1	3	ObjectsReturnedCount
1	4	IdleObjectsSize

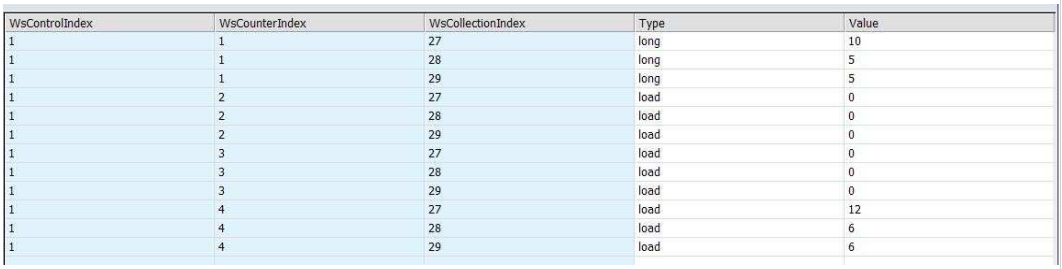
The **collection table** specifies all the MBeans that match the regular expression in the XML file.

The **control table** gives the expression and the counter mode specified in the XML file.

The **counter table** gives the list of the matched counters. If it is automatic, it lists all the counters of the MBeans listed in the collection table. If it is manual, it displays the counters configured in the XML file.

Viewing the metrics (4 of 4)

- Data table



WsControlIndex	WsCounterIndex	WsCollectionIndex	Type	Value
1	1	27	long	10
1	1	28	long	5
1	1	29	long	5
1	2	27	load	0
1	2	28	load	0
1	2	29	load	0
1	3	27	load	0
1	3	28	load	0
1	3	29	load	0
1	4	27	load	12
1	4	28	load	6
1	4	29	load	6

The **data table** gives the values of different statistics.

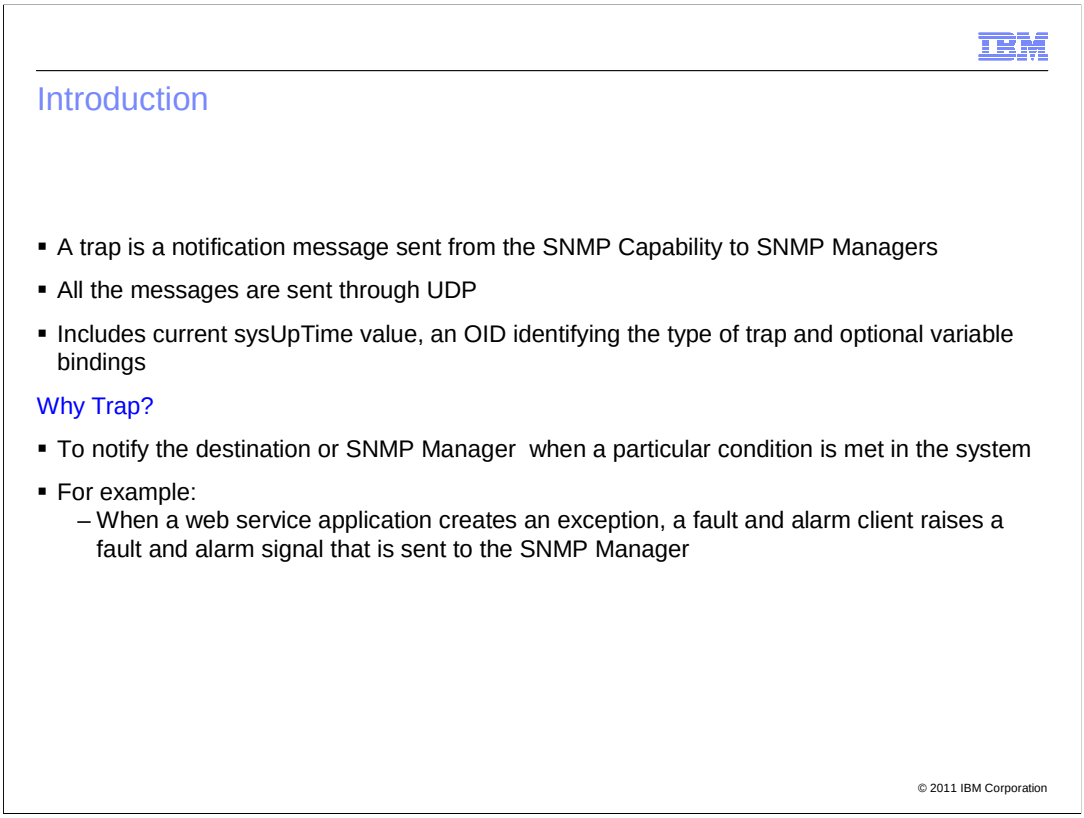
The data table indexes into the previous three tables. It uses the control index, counter index, and collection index to specify the MBean and the attribute, whose value is displayed.

The type **column** gives the three types of statistics - long, load, and stat.

long is an integer

stat gives you an average value, and

load is a range value



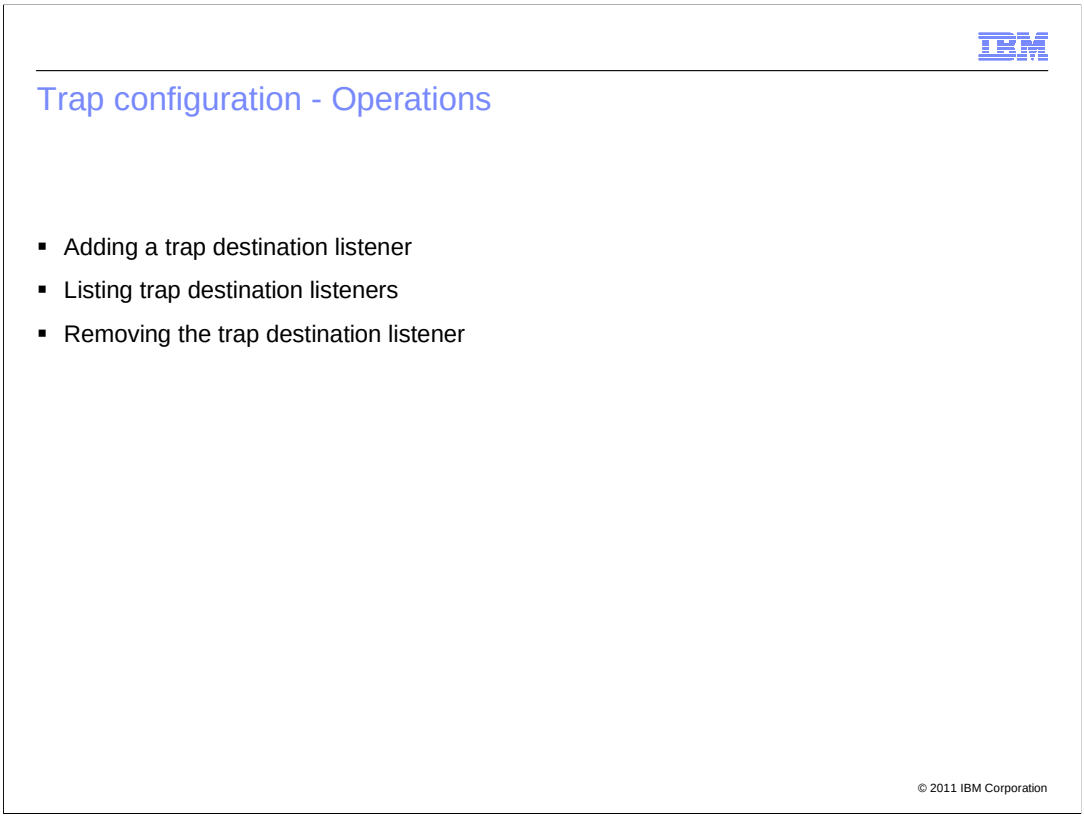
A trap is a notification message sent from the SNMP Capability to the SNMP Managers.

All these messages are sent through UDP to indicate any system status to the SNMP managers.

The trap messages sent from the SNMP capability to the SNMP managers include system time values and OID identifying the type of trap and optional variable bindings to indicate what kind of messages are being sent.

A trap is required to notify the destination or SNMP Manager when a particular condition is met in the system.

For example, when a web service application creates an exception, a fault and alarm client raises a fault/alarm signal that is sent to the SNMP Manager.

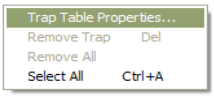


There are three basic operations on trap configuration:

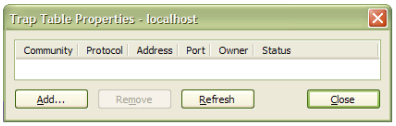
- Adding a trap destination listener
- Listing trap destination listeners
- Removing the trap destination listener

Adding trap destination listener to SNMP Capability (1 of 3)

- The trap destination listener can be added using an SNMP-based software.
  For example:
  - Using MIB explorer (one of the SNMP-based monitoring software)
    - Configure the MIB explorer to connect to SNMP Capability.
    - Click the **Traps** tab on the left bottom pane of the MIB Explorer.
    - Right click the window, and select **Trap Table Properties** in the pop up menu.



- Click **Add...** to add a new Trap Destination listener.



- Enter the details and click **Add**.

There are three ways by which you can add a trap destination listener to the SNMP Capability.

The first method is to use an SNMP-based software.

For example, in a MIB explorer:

1. Click the **Traps** tab on the left bottom pane.

2. Right-click the window that is displayed, and select **Trap Table Properties** in the pop up menu.

3. Click **Add...** to add a new Trap Destination listener.

4. Enter the host name of the trap destination listener, the port where the SNMP-based manager is running, and the community name.

5. Then, click **Add**. The trap destination listener is added to the SNMP capability list.

Adding trap destination listener to SNMP Capability (2 of 3)

- Administrator can modify the trapConfig.xml to add SNMP trap listener destinations when the SNMP Capability is stopped. If the xml file is modified when the SNMP Capability is running, the capability should be restarted.
- A trap can also be added by running this command at the SNMP Capability command prompt
- trapListenerConfig -option <name> <community> <version> <host/ipaddress> <portnum> [<protocol/transport> <status> <index>]
  - option – option can be add, remove or list
  - name – any unique name to identify this Trap Listener
  - community – SNMP community name, for example: public
  - version – SNMP version number. supported values [1, 2, 2c]
  - host/ipaddress – host name or IP address of the trap listener machine
  - portnum – port number where the SNMP listener is running

You can also add a trap destination listener through a command line. Type the command given in this slide and specify the parameters required.

The parameters are:

- option - option can be add, remove or list
- name - any unique name to identify this Trap Listener
- community - SNMP community name, for example: public
- version - SNMP version number. supported values [1, 2, 2c]
- host / IP address - host name or IP address of the trap listener machine
- port number - port number where the SNMP listener is running

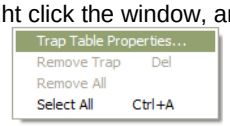
Adding trap destination listener to SNMP Capability (3 of 3)

- protocol/transport – optional; Integer value
- Default – 1 (ip)
  – 2 (ipx)
- Status – value indicating the status of the table entry
  - 1 – active
  - 2 – notInService
  - 3 – notReady
  - 4 – createAndGo
  - 5 – createAndWait
  - 6 – destroy
- Index – index of the table row; appended at the end of the table

This slide has details of the parameters that are required to add the trap destination listener through the command line.

Listing trap destination listeners

- Trap destination listeners can be viewed by using an SNMP-based monitoring software. For example - Using MIB explorer:
  - Click the **Traps** tab at the left bottom pane of the MIB explorer.
  - Right click the window, and select **Trap Table Properties** in the pop-up menu



Trap destination listeners can also be viewed by running this command at the SNMP Capability command prompt:

trapListenerConfig -list

```
================================================================================
================**********ACTIVE TRAP DESTINATION LISTENERS**********================
================================================================================

NAME          COMMUNITY      VERSION   HOSTNAME/IP       PORTNUMBER     TRANSPORT
--------------------------------------------------------------------------------

abcddomain    public         1         X.X.X.X           163            UDP
wxyzdomain    public         1         X.X.X.X           163            UDP
================================================================================
```


To list or view the trap destination listeners through the MIB explorer:

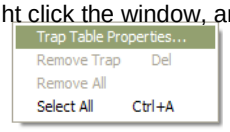
1. Click the **Traps** tab on the left bottom pane of the MIB explorer.

2. Right-click the window, and select **Trap Table Properties** to see the registered trap destination listeners.

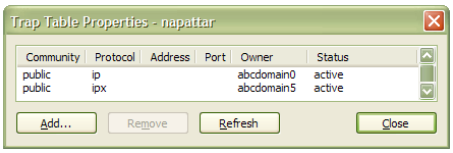
You can also view the available trap destination listeners through the command line by executing the command - *trapListenerConfig -list*

Removing trap destination listener entry (1 of 2)

- Trap destination listeners can be removed by using an SNMP-based monitoring software. For example - Using MIB explorer:
  - Click the **Traps** tab on the left bottom pane of the MIB Explorer.
  - Right click the window, and select **Trap Table Properties** in the pop-up menu.



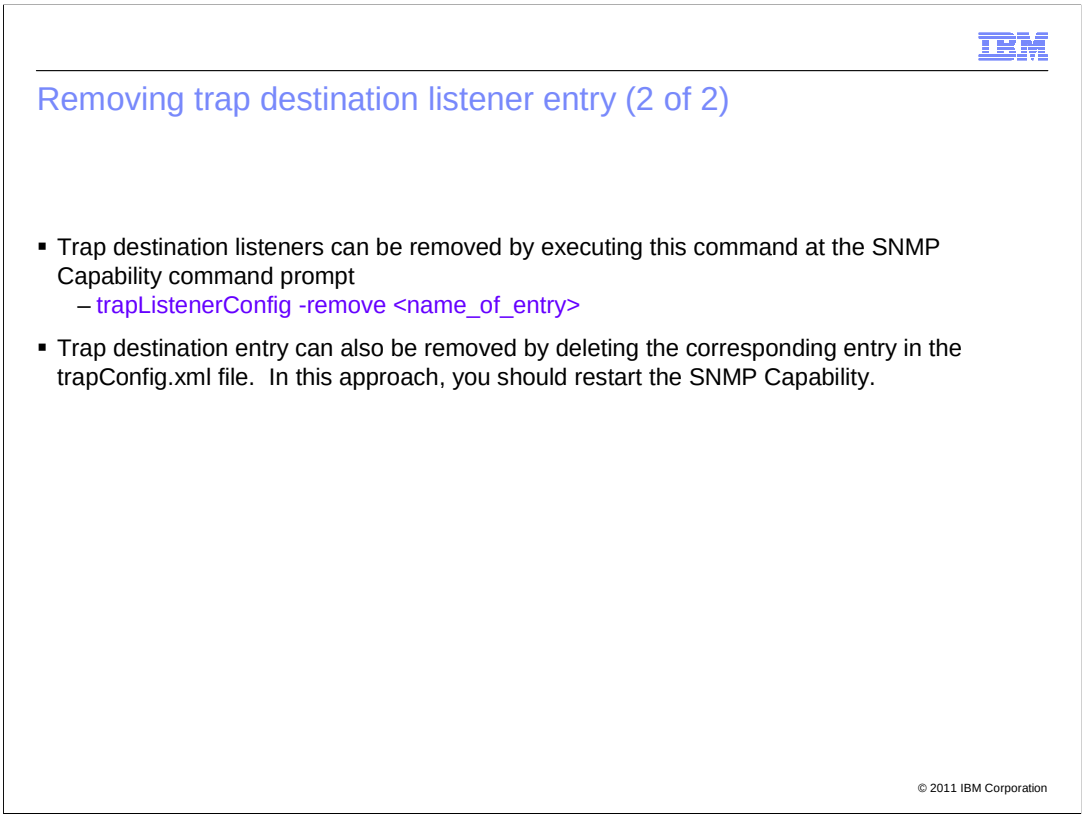
The traps are listed.



Select an entry and click **Remove** to remove the listener.

To remove a trap destination listener entry from the SNMP capability database:

1. Click the **Traps** tab on the left bottom pane of the MIB explorer.
2. Right-click the window, and select **Trap Table Properties**.
3. From the listed traps, select the entry to be removed, and then click **Remove**.



Trap destination listeners can also be removed from the SNMP capability through the command line by using the command - *trapListenerConfig -remove <name_of_entry>*.

You should specify the name of the entry to be removed.

Another method to remove the trap destination entry is by deleting the corresponding entry in the trapConfig.xml file. You should restart the SNMP Capability after this.

Processing SNMP traps

- SNMP Capability waits for the JMX notification generated from WebSphere Application Server
- The JMX notifications are triggered by Fault and Alarm client of the ICSE components in case of error conditions
- SNMP Capability handles two types of JMX notification events

NOTIFICATION EVENT	TYPE / ID	DESCRIPTION
FAULT	ibm.ims.fault	This type of event is generated whenever there is a fault generated during Web service operation.
ALARM	ibm.ims.alarm	This type of event is triggered whenever there is an alarm generated from Web service operations.

SNMP Capability can generate SNMP V1 or V2 version traps depending on the type of the registered listener

The fault and alarm client in the TWSS web services raises ibm.ims.fault and ibm.ims.alarm type of events.

SNMP Capability listens for these particular events. Whenever there is an event by this ID, SNMP Capability generates a trap to a particular trap destination listener, which is already registered.

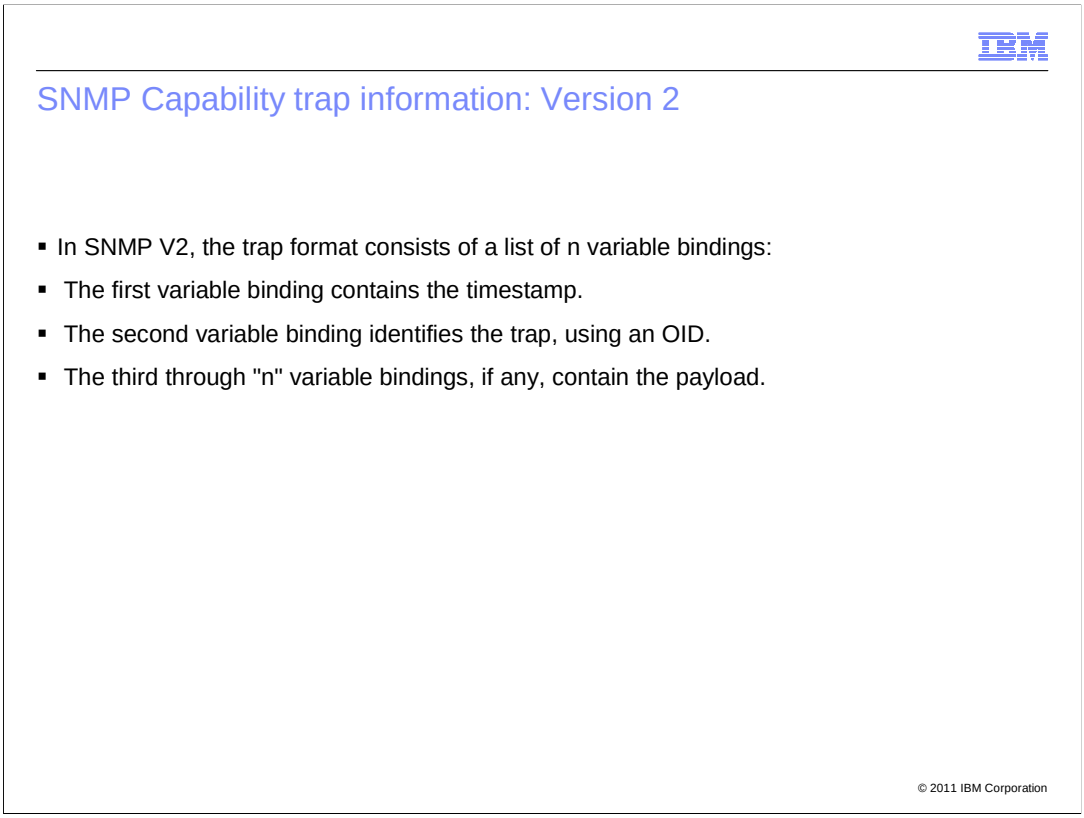
SNMP Capability trap information: Version 1

SNMP version V1 Trap PDU has this information in the PDU.

PDU Type	An integer value indicating the PDU type, which is 4 for a Trap-PDU message
Enterprise	An object identifier for a group, which indicates the type of object that generated the trap
Agent Address	The IP address of the SNMP Capability that generated the trap
Generic Trap Code	A code value specifying a predefined “generic” trap type
Specific Trap Code	A code value indicating an implementation-specific trap type
T**ime Stamp**	The amount of time since the SNMP entity sending this message last initialized or reinitialized; used to time stamp traps for logging purposes
Variable Bindings	A set of name-value pairs identifying the MIB objects in the PDU; see the general message format topic for more on these bindings

Listed here are the fields that are available in the trap received for SNMP version v1.

- PDU Type - is 4 for a Trap-PDU message
- Enterprise - An object identifier for a group
- Agent Address - The IP address of the SNMP Capability that generated the trap
- Generic Trap Code - A code value specifying a predefined number
- Specific Trap Code - A code value indicating an implementation-specific trap type
- Time Stamp - The time when the trap was generated.
- Variable Bindings – gives the details of the error message.



SNMP version 2 trap information is similar to version 1, but the trap format consists variable bindings only:

- The first variable binding contains the timestamp.
- The second variable binding identifies the trap, using an OID.
- The third through "n" variable bindings, if any, contain the payload.

Securing SNMP data

SNMP Capability secures user name, password, and community names by encrypting the values for the corresponding attributes. It encrypts these attributes specified in the agentConfig/jmxConfig xml files.

user	SNMP Capability encrypts the ‘user’ attribute value specified in the jmxConfig configuration file.
Password:	SNMP Capability encrypts any attribute name in the jmxConfig configuration file, which ends with Password.
password	SNMP Capability encrypts any number of attributes by name ‘password’.

Whenever an SNNMP Capability is deployed, there is need to secure the data.

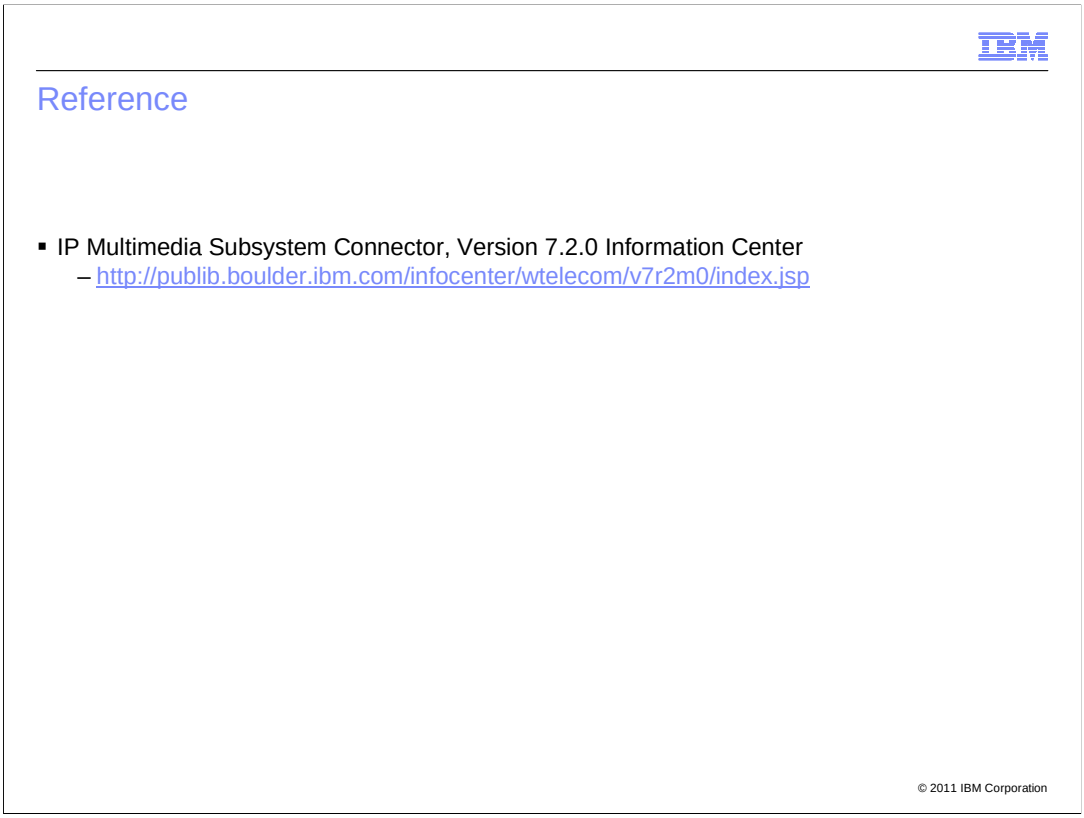
SNMP Capability secures user name, password, and community names specified in the agentConfig/jmxConfig xml files by encrypting the values for the corresponding attributes.

Encrypted attributes are:

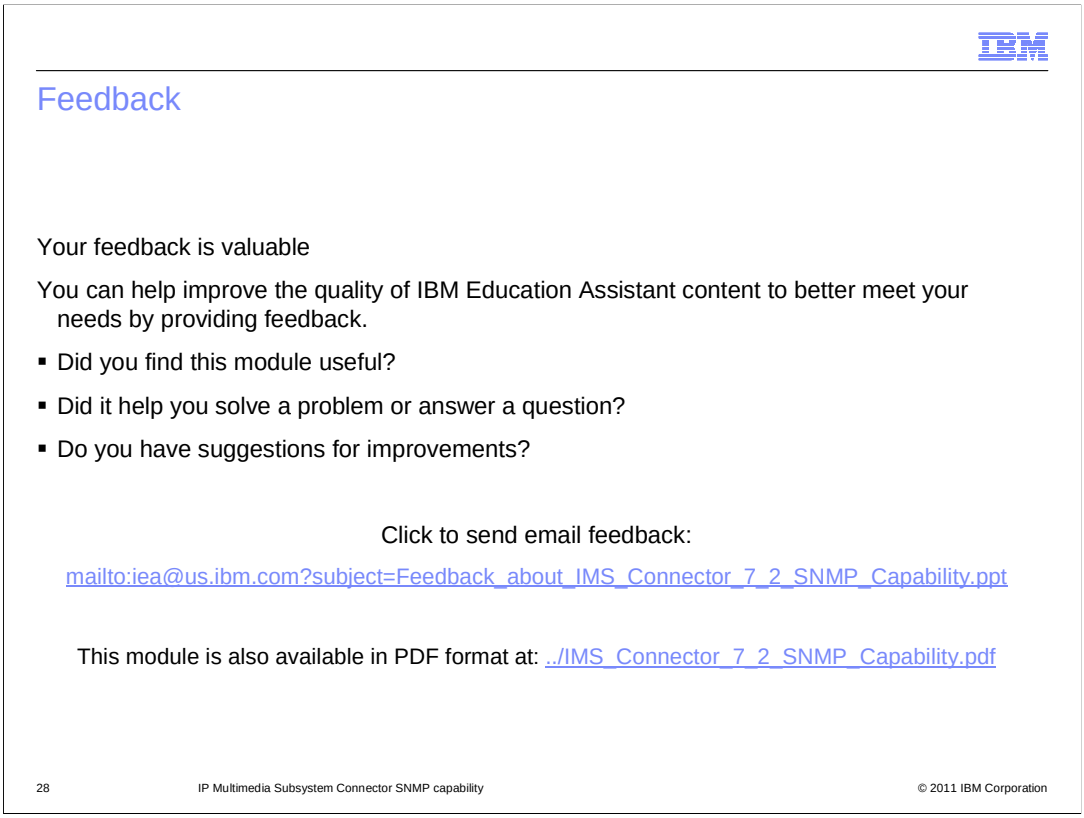
- user - SNMP capability encrypts the ‘user’ attribute value specified in the jmxConfig configuration file.

- Password: - SNMP capability encrypts any attribute name in the jmxConfig configuration file, which ends with Password.

- password - SNMP capability encrypts any number of attributes by name ‘password’.



For more information, see the Information Center (http://publib.boulder.ibm.com/infocenter/wtelecom/v7r2m0/index.jsp).



You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, disclaimer, and copyright information