TTM 4128 Exam May 19th 2009

Enclosed :
ASN.1 Encoding Scheme
Shortened version of RFC 1905

Task 1: SNMP v1 and 2 (10%)
Please provide short answers.

1.1 Explain shortly the application area of SNMP.

SNMP are used as a platform for management of networked components such as end-systems and network components such as PCs, LAPTOPS, Printers, Routers, switches etc. Management comprises the setting and retrieval of state data as well as data used in the operation of the systems.

1.2 List five among the variety of kind of information objects that are identified by use of object identifiers defined by the Management Information Tree.

The concept information object is not used in the book or in the lecturers. The concept managed object is however used. Managed objects in SNMP are scalar information objects with more than one variable defined by a macro. In SNMP Managed Object Types and identified by object identifiers. Object identifiers are also used for the identification of managed object instances, but with additional help of port numbering and indexes. In the system object the variable System Object ID is used to identify the specification of the vendor equipment. What is asked about are other information objects. In SNMP v2 there are many information objects defined by macros in addition to SNMPv1 objects. So we have:

Managed Object Identities
Managed Object Groups
Managed Objects Types
Managed Object Instances
Managed Equipment Types
Module Identity
Notification Groups
Notification Type

1.3 We are considering components of the organization, information and communication models of the SNMP management model. Define the main components defined in the various models and give a simple example that explains how components are used in a real executing network management system.

Manger: Controls the network management functionality.
Agent: A representative of the manager in the domain of the managed system
SNMP protocol: The protocol used in the communication between manger and agent
MDB (Management data base): A database containing the instance values of the managed objects retrieved
MOIV (Managed Object Instance values): Managed object instance values in the agent
MIB (Management Information Base): The managed object types

A manager decides to retrieve a managed object instance value from a managed system. The manger must find:
- the IP address where the agent controlling the managed object value is
- the Object identifier of the managed object type

The manger sets the protocol fields defining the message type, community etc as well as the Varbind elements of the SNMP PDU defining object identifier and syntax and an ASN.1 encoded SNMP PDU is sent to the agent .

The agent receives the SNMP message, decodes it and by using MIB it can identify which managed object the manager is addressing. The agent will access the MOIV, and will set the Varbind elements of the SNMP PDU to be returned. An ASN.1 encoded SNMP message is returned to the Manger.

The Manger decodes the SNMP message and will identify the managed object received by using the MIB. The MDB van then be updated,

Task 2: ASN.1 encoding (10%)

2.1 What is the type field BER encoding of the ASN.1 type SEQUENCE? What is the total BER encoding of an instance of INTEGER with value 1?
SEQUENCE: UNIVERSAL 16 constructed = 00110000 = 30 hex
INTEGER: UNIVERSAL 2 = 00000010 = 02 hex.
INTEGER with value 1 is 020101 hex

2.2 What is the BER encoding of an instance of OBJECT IDENTIFIER with value {1 3 6 1}?
{1 3 6 1} is encoded 06 03 43 06 01 hex

2.3 What is the total BER encoding of an instance the following ASN.1 defined type ProtocolMessage, where

ProtocolMessage ::=
SEQUENCE {community INTEGER, id OBJECT IDENTIFIER, ApplicationSyntax}
with ApplicationSyntax::= CHOICE{ INTEGER , OBJECT IDENTIFIER }
and where:
community INTEGER ::= 1
id OBJECT IDENTIFIER ::= {1 3 6 1}
ApplicationSyntax in this instance is INTEGER with value 1
Bottom up and counting gives:
Instance of application syntax from 2.1: ***INTEGER with value 1 is 02 01 01 hex***
Instance id from 2.2: ***{1 3 6 1} is encoded as 06 03 43 06 01 hex***
Community from 2.1 ***INTEGER with value 1 is 02 01 01 hex***
Total length is 11 octets = ***0B hex***.
From 2.1: ***SEQUENCE type encoding is 30 hex***
The result is: ***30 0B 02 01 01 06 03 43 06 01 02 01 01***

Task 3: Tables as managed objects (10%)
Please provide short answers.

3.1 What are Tables used for in SNMP?
Tables are used to represent sets of managed objects that are repeated. Examples are routing tables, interface tables, TCP connection Tables.

3.2 What are the five basic concepts used to define the structure of Tables in SNMPv1?
Basic concepts are: Table, Entry, Columnar Objects, Rows and Index

Tabular Representation of Aggregate Objects

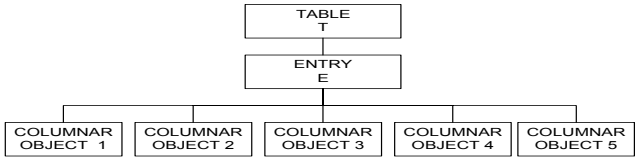
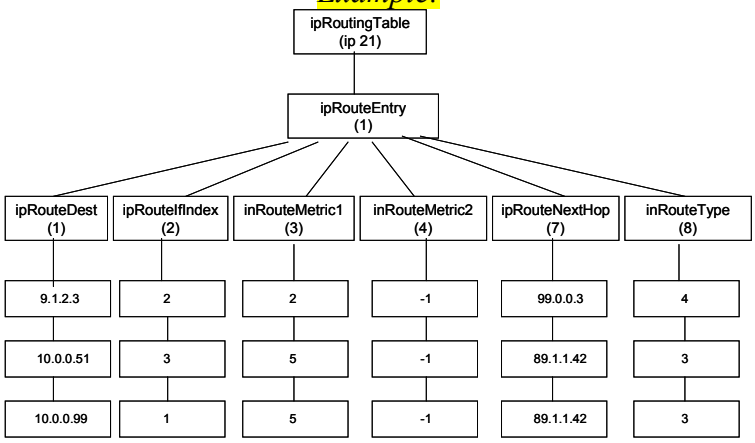


Figure 4.22(a) Multiple Instance Managed Object

- The objects *TABLE T* and *ENTRY E* are logical objects.
  - They define the grouping and are not accessible
- Columnar objects are objects that represent the attributes and hence are accessible
- Each instance of *E* is a row of columnar objects 1 through 5
- Multiple instances of *E* are represented by multiple rows

Example:



3.3 Explain shortly the purpose and role of the SNMPv2 Textual Convention RowStatus?

```
RowStatus ::= TEXTUAL-CONVENTION
   STATUS       current
   DESCRIPTION
         " ..........."
   SYNTAX       INTEGER {
                   active(1),
```

```
        notInService(2),
        notReady(3),
        createAndGo(4),
        createAndWait(5),
        destroy(6)
    }
```

The purpose of RowStatus is to be able to create and delete lines of the Table. By using RowStatus as a columnar object of a table this is possible.

Task 4: SNMP protocol v2 (10%)

4.1 Explain what a notification is. A short version of RFC 1905 is given as Attachment. Explain short the encoding of instances of the type Varbind for SNMP PDUs not containing notifications.

A notification is a messages sent from the agent to the manger based on the initiative of the agent.

```
VarBind ::=
    SEQUENCE {
        name                ObjectName,
        CHOICE {
            value               ObjectSyntax,
            unSpecified         NULL,
            noSuchObject[0]     IMPLICIT NULL,
            noSuchInstance[1]   IMPLICIT NULL,
            endOfMibView[2]     IMPLICIT NULL }
    }
```

A Varbind element correspond to a managed object instance and is defined by two types: The first type ObjectName is the type ObjectIdentifier and identifies the object type in the MIB tree. The second type is in normal cases ObjectSyntax which corresponds to the type of the managed Object. So normally one Varbind instance is encoded as

ObjectName type, ObjectName length, ObjectName value,
ObjectSyntax Type, Object Syntax length, Object Syntax Value.

For Varbind elements that does not carry a Object Syntax Value, for example when a managed object instance value is requested from the manger the encoding Null is used. For Varbind elements from the agent in non-normal cases such as noSuchObject, noSuchInstance, and endOfMibView a context specific tag implicit Null is used.

4.2 An example of the use of NOTIFICATION-TYPE MACRO is given as follows:

```
linkUp NOTIFICATION-TYPE
        OBJECTS          {ifIndex}
         STATUS          current
         DESCRIPTION     " ……."
::= {snmpTraps  4}
```

Explain the various elements of this definition.

linkUp is an Object Identifier for the trap in the MIB tree and is located as {snmpTraps 4}

OBJECT lists the set of Object Identifiers that shall be carried on an instance of this trap from the agent

STATUS defines the validity of this notification definition. Current defines this definition as the current version notification

4.3 An instance of linkUp is sent from an Agent to a Manager. In this case the Varbind list will carry 4 TLV elements. The first TLV element is TLV for sysUpTime. What are the 3 other elements?

PD Type	RequestID	Error Status	Error Index	VarBind 1 sysUpTime OID	VarBind 1 sysUpTime value	VarBind 2 snmpTrapOID	VarBind 2 notification value	.. .

Varbind element 1: TLV for sysUpTime from System Object.
Varbind element 2: TLV for linkup
Varbind element 3: TLV for ifIndex
Varbind element 4: TLV for snmpTrapEnterprise, i.e. System Object ID from System Object

Task 5 CIM and CIM /WBEM. (20%)
Please provide short answers.

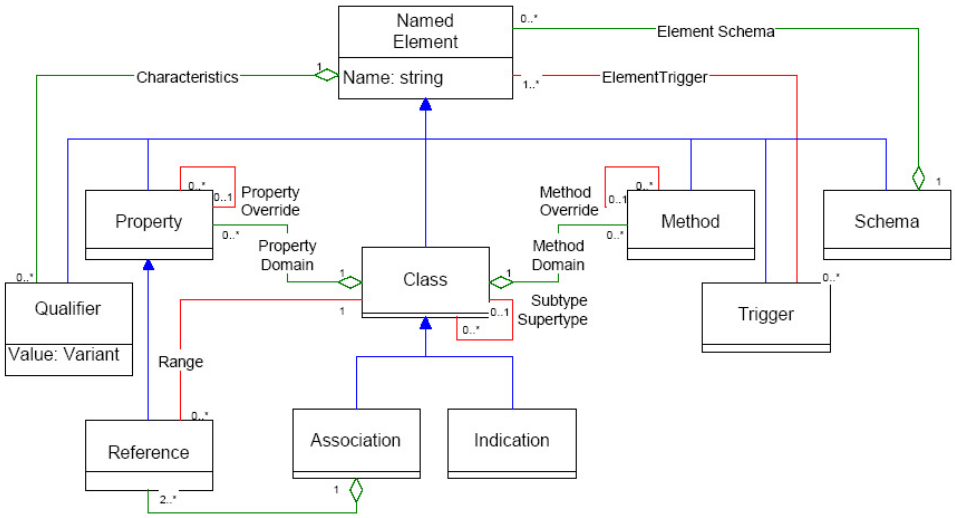
5.1. CIM.
a) What is CIM and what are the CIM components?
b) What is UML used for in the CIM standard?
c) What are "Named Elements" in a CIM representation? Explain the usage.

a) CIM provides the common conceptual definitions for the managed objects, their properties, and relationships. It defines how managed objects in a network/environment are represented as a common set of objects and relationships between them. CIM is a data model not an implementation.

The CIM components are CIM specification, CIM Schema and CIM Extention schema.

b) UML Class diagram (its meta model and the graphical representation) is used to express the CIM specifications. UML is one way of expressing CIM; the other ways are XML and MOF

c) As shown in the main meta schema of the CIM standard, every element in the CIM model is a Named Element.

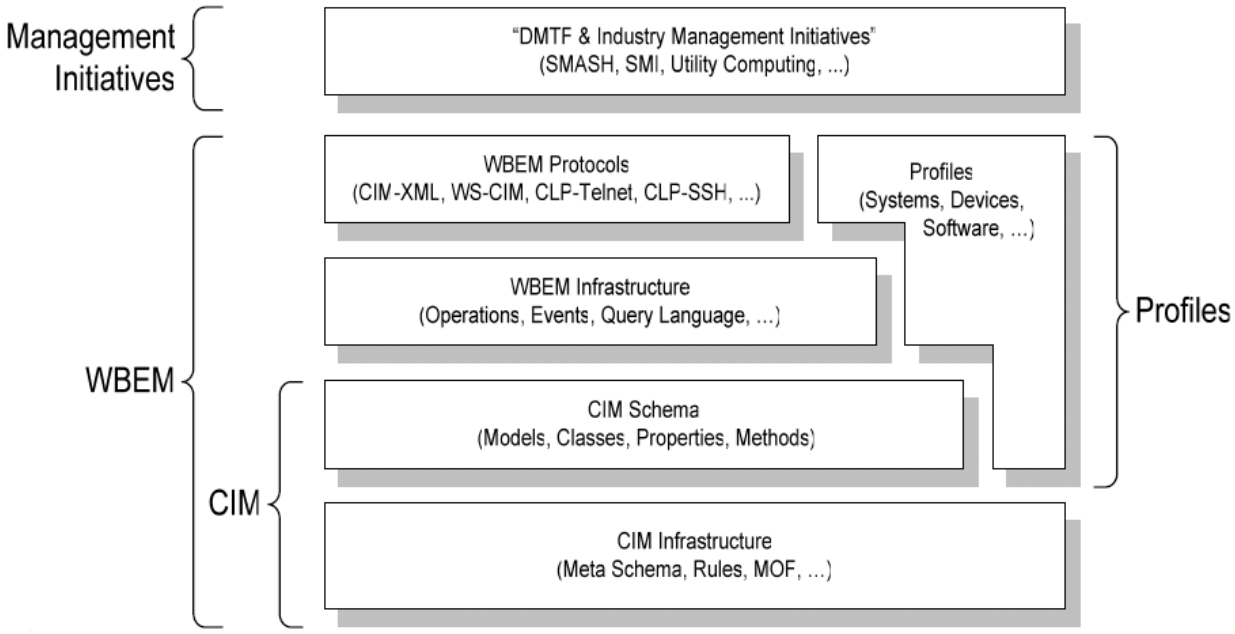


5.2. CIM/WBEM.

a) Explain the role of CIM in WBEM.

b) What is the difference between cimXML and XML-CIM?

a) WBEM is an industry initiative and umbrella of network management standards and model. The CIM standard is used as the only data and information model in WBEM.



b) CIM-XML is a WBEM protocol that uses XML over HTTP to exchange Common Information Model (CIM) information. CIM-XML uses xmlCIM as the payload and HTTP as the transport. xmlCIM is a standard way to represent CIM data using the Extensible Markup Language (XML). Enables CIM declarations (Classes, Instances, and Qualifiers) and messages to be described in an XML.

5.3 CIM related to SNMP

Discuss the relationship between some of the main concepts of SNMP and some of the main concepts of CIM and CIM/WBEM.

SNMP is a network management protocol while CIM is a conceptual information model. An SNMP implementation can constitute a management platform that is capable of handling both management information data and the communication between managers and agents. A CIM implementation, on the other hand, only handles how the management information data are structured and specified. Following is a list of the main SNMP concepts and their corresponding or related CIM/CIM-related WBEM concepts.

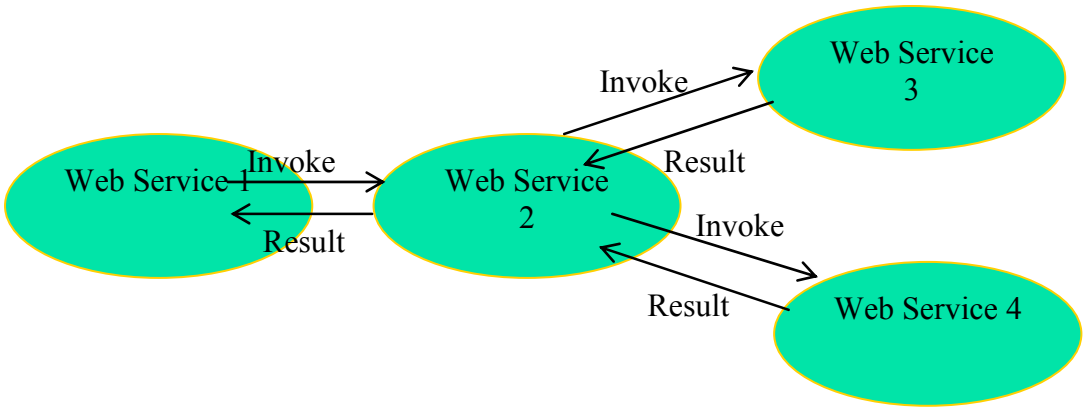
SNMP concept	CIM concepts	Clarification
MIB	CIM Schema	A CIM Schema defines a set of managed objects – as classes
MIB tree	-	No such concept in the CIM terminology, as managed objects are modeled as classes in a class hierarchy
OID	CIM Instances, CIM Objects	A CIM class (instance and object) is the corresponding entity to an SNMP OID
SMI	CIM Meta schema, MOF, CIM Specification	The CIM terminology of the
get-request	-	No specific requests nor traps exist in CIM
MIB group	-	Grouping of managed objects is not specifically provided in the CIM as in the SNMP MIB definitions
MIB module	Core Model, Common Models, CIM Extension Schemas	CIM information is structured in core model, common models, and extension schemas
Trap	-	No specific requests nor traps exist in CIM
SNMP protocol	CIM-XML	CIM-XML is the WBEM protocol that provides the means for exchanging CIM information
Agent	-	In an implementation of a CIM/WBEM a client application would correspond to the SNMP agent object, thus no specific entity in the architectural model
Manager	CIMOM	CIMOM in the WBEM corresponds to the SNMP manager object

Task 6. Web services (10%)
Please provide short answers.

6.1. What is Web service?

A Web service

- *Can be any computational functionality that can be found and invoked over any network (e.g. the Internet).*
- *Represents a self-describing, self-contained application.*
- *Can be designed to be used by other programs or applications rather than humans.*
- *Can be accessed via transparent Internet protocols like http, ftp etc.*
- *Can be mixed and matched with other Web Services in a value chain.*
- *Is described by WSDL (Web Service Description Language)*
- *Uses SOAP (Simple Object Access Protocol) for information exchange*



6.2. How can Web service be used in network management? Sketch and explain one solution.

In Network management, the model Manager- Agent is broadly used. A Manager manages several Managed object via Agents. In the OSI systems management (OSI-SM) using the Common Management Information Service/ Protocol (CMIS/P) the Managed object supports full inheritance, provide interface for both Get, Set, Action, Create, Delete and advanced event system. The Web service concept is quite suitable to implement the Managed-object exposing interface containing methods for Get, Set, Action, Create, Delete.

One solution is to add embedded web-service based agents to network components. Data is pushed by the management agents in managed objects to the manager applications. WS are used to utilize the network management applications and their remote interfaces, e.g. remotely apply maintenance, provisioning, monitoring, configuration, etc. activities via a SOAP interface.

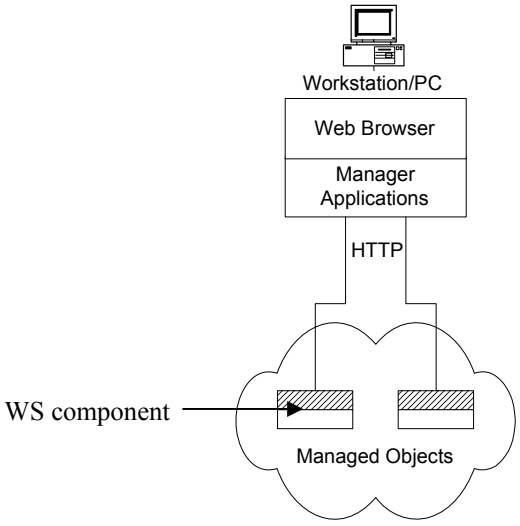


Figure 14.3 Embedded WBM Configuration

Task 7. Design of a network management application system (20%)

A network management application to manage network resources remotely shall be designed. The application will consist of two parts. Part 1 is running on a PC, and Part 2 is running on a Managed Node (storage devices, CPUs, databases, etc.)

The communication between your PC and the node is only possible via HTTP protocol. See the figure. Network management requests/responses are carried as HTTP payload.

Assume also there are a firewall and a proxy between the PC and the Node. The firewall prevents any remote connectivity (ssh, vnc, telnet, etc.), while the proxy does not allow any network management PDUs at the application protocol level. This can for instance be realized by filtering mechanisms at certain port numbers.

To develop your network management application, you can make use of semantic web, web services, and a network management standard such as SNMP or CIM/WBEM.

Based on all these assumptions defined above, the network management application system shall be described.

7.1. Describe your proposed solution regarding components, connections, protocols, used development platforms, servers, etc.

A possible implementation/realization of such an application would contain the following components:

- *Web server at the Managed Node (e.g. Tomcat web server*
- *Web Service applications: an application to create the management requests at the PC and an application that will act as a service provider at the managed node (both applications are developed using Apache Axis and Apache Ant for instance)*
- *SNMP protocol platform to be used as an agent software at the managed node (e.g. net-snmp tool)*
- *Semantic Web platform at both the PC and the managed node (e.g. JENA)*
- *An ontology defining the object identifications of the managed resources*

7.2 Assume the proxy is also applying content-filtering, e.g. not allowing http PDUs that contains any standard object identification for resource management. How can you overcome this obstacle?

This can be overcome by defining ontology for synonyms or aliases for the object identifications of the resources to be managed. This ontology will be used at one side of the communication to map (mapping here is similar to crypting) the requested object identifications to different names that will bypass the content-filtering at the proxy. The same ontology would be used at the other side of the communication to obtain the original object identifications.

7.3 What could ontology reasoning be used for in your network management application?

Managed resources will be classes in ontologies. To define aliases of a class we need to define subclasses to that class. A reasoner could be used to provide the needed restrictions for subclasses of class in terms of its properties, i.e. to ensure that subclasses preserve the same properties as that class.

7.4 Assume that SOAP traffic is only allowed in one direction from your PC to the Managed Node. How is it possible to obtain regular updates from Managed Node?

Regular updates can be implemented as a stand-alone application that triggers traps from the managed node (for instance by issuing a net-snmp trap command). These traps can then be collected in a dynamically generated web page (using pushing mechanism for instance). In this case we don't use SOAP.

Task 8. Autonomic/Adaptive Systems (10%)

Please provide short answers.

8.1 List the main Self-* properties of autonomic systems.

- ❑ *Self-Configuring*
- ❑ *Self-Healing*
- ❑ *Self-Optimizing*
- ❑ *Self-Protecting*

8.2 Explain in more detail two of such Self-* properties. Give examples of platforms/systems that support autonomic/adaptive systems.

- Self-Configuring

Characteristics that enable systems to adapt to changing conditions by changing their own configurations. Functionality that allows the addition and removal of components or resources without service disruption

- Self-Healing

Capacity to recognize and diagnose deviations from normal conditions and take action to normalize them. Capability to proactively circumvent issues that could cause service disruptions

- Self-Optimizing

Ability of the system to monitor its state and performance and proactively tune itself to respond to environmental stimuli

-Self-Protecting

Incorporation of intelligence to recognize and circumvent security threats. Facility of a system to protect itself from physical harm, such as excessive heat or motion

Example such autonomic platforms/systems are: UPnP, Service Discovery protocol, IBM Autonomic Management protocols/platform, TAPAS, Ad-Hoc routing protocols, etc.