UNIT 2: BASIC FOUNDATION – STANDARDS, MODELS & LANGUAGE

NETWORK MANAGEMENT STANDARDS

OSI/CMIP

- International standard (ISO/OSI)
- Management of data communications networks--LAN & WAN
- Deals with all 7 layers
- Object oriented
- Well structured & layered
- Consumes large resource in implementation
- The OSI management protocol standard is CMIP (Common Management Information Protocol) , & has built-in services ,CMIS (Common Management Information Service) that specify the basic services needed to perform the various functions

SNMP/Internet

- Industry standard (IETF)
- Originally intended for management of Internet components, currently adopted for WAN & telecommunication systems
- Easy to implement
- Most widely implemented

TMN

- International standard (ITU-T)
- Management of telecommunications network
- Based on OSI network management framework
- Addresses both network & administrative aspects of management

IEEE

- IEEE standards adopted internationally
- Addresses management of LANs & MANs
- Adopts OSI standards significantly
- Deals with first 2 layers of the OSI reference model

Web Based Management

- This is based on using Web technology, a web server for the management system and web browsers for network management stations
- Web Based Enterprise Management (WBEM)
- Java Management Extensions (JMX)
- DMTF (Desktop Management Task Force) is developing specifications for WBEM.
- JMX is based on a special subset of Java applets developed by Sun microsystems that runs in the network components

NETWORK MANAGEMENT MODEL

- OSI network management architecture model comprises of 4 models: organization model, information model, communication model & functional model (Figure: 3.1).
- The functional model deals with the user-oriented requirements of network management.
- The information model deals with the structure & organization of management information.
- The communication model has 3 components: management application processes that function in the application layer, layer management between layers and layer operation within the layers.
- The organization model describes the components of a network management system, their functions and their infrastructure.

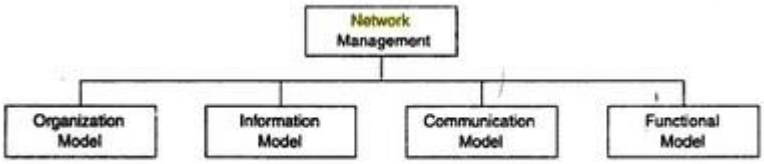

Figure 3.1 OSI Network Management Model

ORGANIZATION MODEL

- The organization model describes the components of network management & their relationships.

Two Tier Network Management Organization Model

- In two tier model (Figure: 3.2), network objects consists of network elements such as hosts, hubs, bridges, routers etc.
- They can be classified into managed & unmanaged objects or elements.
- The managed elements have a management process running in them called an agent.
- The manager manages the managed element.
- There is a database in the manager but not in the agent.
- The manager queries the agent & receives management data, processes it & stores it in its database.

Three Tier Network Management Organization Model

- In 3 tier model, the intermediate layer acts as both agent & manager (Figure: 3.3),
- As manager, it collects data from the network elements, processes it & stores the results in its database.
- As agent, it transmits information to the top-level manager.

Network Management Model with MoM

- Network domains can be managed locally and a global view of the networks can be monitored by a MoM (Manager of managers).
- This configuration uses an enterprise network management system & is applicable to organizations with sites distributed across cities (Fi

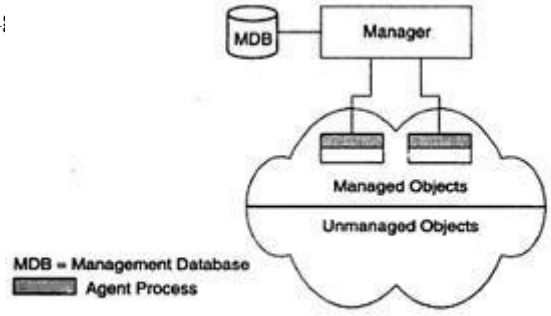

Figure 3.2 Two-Tier Network Management Organization Model

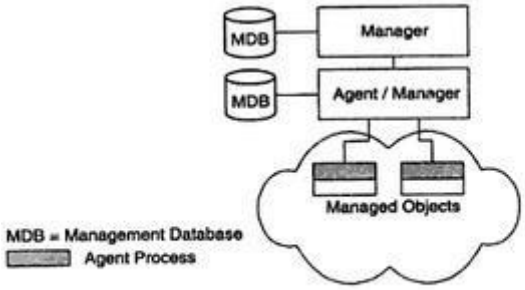

Figure 3.3 Three-Tier Network Management Organization Model

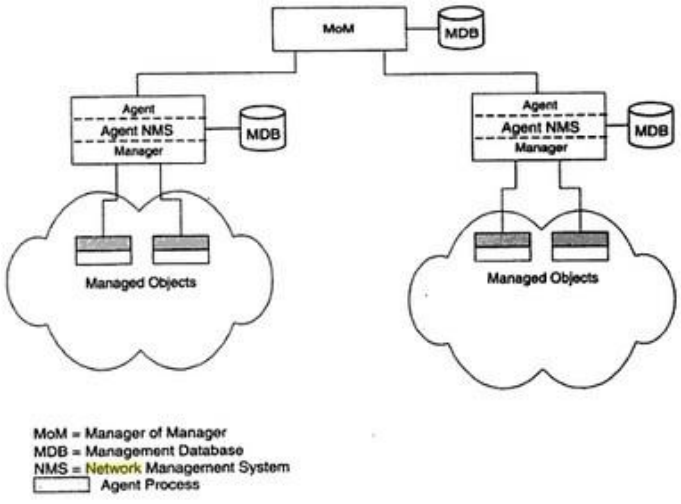

Figure 3.4 Network Management Organization Model with MoM

INFORMATION MODEL

- An information model is concerned with the structure & the storage of information (Figure: 3.6).
- Information on network components is passed between the agent & management processes.
- The information model specifies the information base to describe managed objects & their relationships.
- The SMI defines the syntax & semantics of management information stored in the MIB.
- The MIB is used by both agent & management processes to store & exchange management information.
- A manager MIB consists of information on all the network components that it manages whereas an agent MIB needs to know only its local information, its MIB view.
- The MDB is a real database & contains the measured or administratively configured value of the elements of the network. On the other hand, the MIB is a virtual database & contains the information necessary for processes to exchange information.

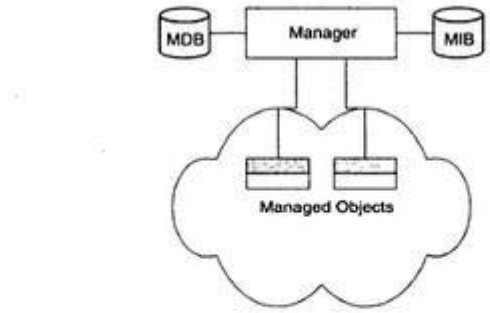

Figure 3.6 Network Configuration with Data and Information Base

COMMUNICATION MODEL

- Management data is communicated between agent & manager processes, as well as between manager processes.
- Three aspects need to be addressed in the communication of information between 2 entities: transport medium of message exchange, message format of communication and the actual message.

Management Communication Model

- In the communication model (Figure: 3.11), the applications in the manager module initiate requests to the agent in the Internet model.
- The agent executes the request on the network elements and returns responses to the manager.
- The notifications/traps are the unsolicited messages such as alarms, generated by the agent.

Management Communication Transfer Protocols

- Figure: 3.12 presents the communication protocol used to transfer information between managed object & managing processes, as well as between management processes.
- The OSI model uses CMIP along with CMIS. Internet uses SNMP for communication.
- OSI uses both connection oriented and connectionless protocols for transportation. Internet uses connectionless UDP/IP protocol to transport messages.
- CMIP & SNMP specifies the management communication protocols for OSI & Internet management respectively.

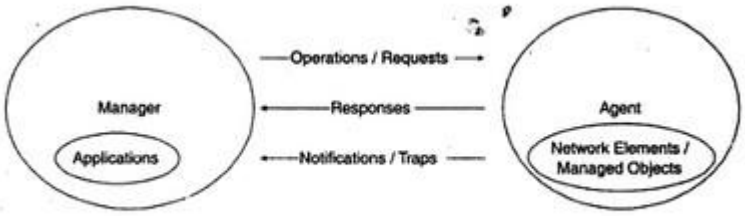

Figure 3.11 Management Communication Model

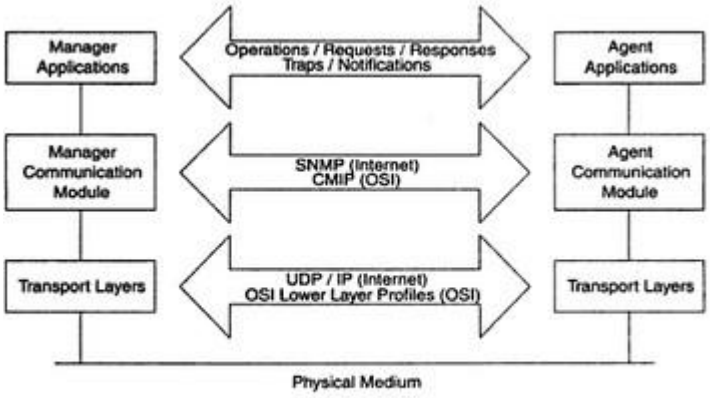

Figure 3.12 Management Communication Transfer Protocols

MANAGEMENT INFORMATION TREES

- Managed objects are uniquely defined by a tree structure specified by the OSI model & are used in the Internet model (Figure: 3.8).
- There is root node & well-defined node underneath each node at different levels.
- Each managed object occupies a node in the tree (e.g. Internet is designated as 1.3.6.1).
- In the OSI model, the managed objects are defined by a containment tree that represents the MIT.
- The root node does not have an explicit designation.
- The iso defines the International Standards Organization and itu defines the International Telecommunications Union.
- The 2 standards organizations are on the first layer & define management of objects under them.
- The joint iso-itu node is for management objects jointly defined by the 2 organizations.

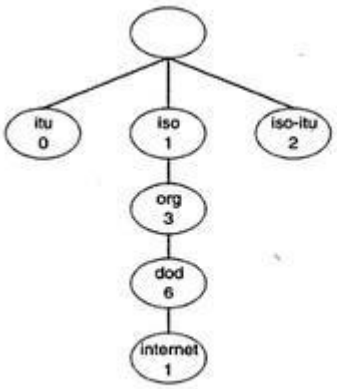

Figure 3.8 OSI Management Information Tree

CONCEPTUAL VIEWS OF MANAGED OBJECTS (INTERNET & OSI PERSPECTIVE)

- A managed object in the Internet model is defined by 5 parameters (Figure: 3.9a):
  - → object identifier & descriptor: unique ID & name for the object type
  - → syntax: used to model the object
  - → access: access privilege o a managed object
  - → status: implementation requirements
  - → definition: textual description of the semantics of object type
- The Internet object model is a scalar model & is easy to understand. In contrast, the OSI perspective of a managed object is complex & has a different set of characteristics
- OSI specifications are object-oriented, and hence a managed object belongs to an object class
- The attribute of an object defines the external perspective of the object
- An OSI managed object has the following characteristics
  - → object class: managed object
  - → attributes: attributes visible at its boundary
  - → operations: operations that can be applied to it
  - → behaviour: behavior exhibited by it in response to an operation
  - → notification: notifications emitted by the object
- Operation in the Internet model is done by get & set commands. Notification is done by response & alarm messages.
- In OSI, we can create & delete objects. These concepts do not exist in the Internet.

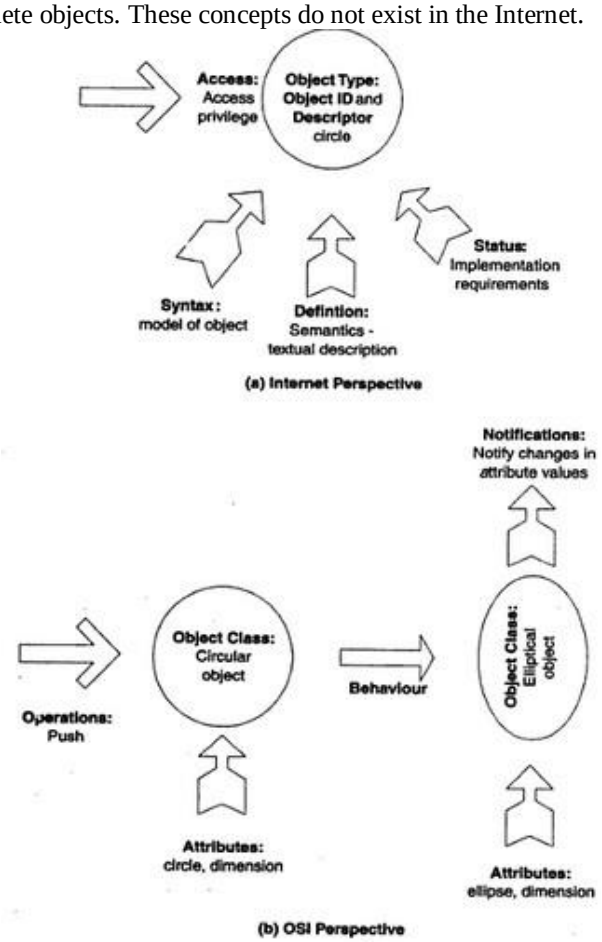

Figure 3.9 Conceptual Views of Managed Object

Internet specifications for the object "Packet Counter".

Characteristics	Example
Object type	PktCounter
Syntax	Counter
Access	Read-Only
Status	Mandatory
Description	Counts number of packets

OSI specifications for the object "Packet Counter".

Characteristics	Example
Object class	Packet Counter
Attributes	Single-valued
Operations	get, set
Behavior	Retrieves or resets values
Notifications	Generates notifications on new values

ASN.1

- ASN.1 stands for Abstract Syntax Notation One.
- This is a formal language developed jointly by CCITT & ISO for use with application layers for data transfer between systems.
- This is also applicable within the system for clearly separating the abstract syntax and the transfer syntax at the presentation layer.
- Abstract syntax is defined as the set of rules used to specify data types and structures for storage of information.
- Transfer syntax represents the set of rules for communicating information between systems.
- Abstract syntax is applicable to the information model and transfer syntax to the communication model
- The algorithm to convert the textual ASN.1 syntax to machine readable code is called BER (Basic Encoding Rules).

ASN.1 CONVENTIONS

- ASN.1 is based on the Backus system & uses the formal syntax language & grammar of the BNF (Backus-Nauer Form) ,which looks like

  <name>::=<definition>

  where the notation <entity> denotes an "entity" and the symbol ::= represents "defined as"

  e.g.: *<BooleanType>::= BOOLEAN*

  <BooleanType>:= TRUE | FALSE

  The definitions on the right side are called primitives

  The format of each line is defined as a production or assignment

  Entities that are all in capital letter such as TRUE and FALSE are called keywords

- A group of assignments makes up an module.

  eg: person-name Person-Name ::=

  {

  first "john"
  middle "T"
  last "smith"

  }

  Here "person-name" is the name of the module which is a data type. "Person-Name" is a module

- Following are 3 constructive mechanisms:

  → alternatives: CHOICE

  → list: SET and SEQUENCE

  → repetition: SET OF and SEQUENCE OF

- ASN.1 definition allows both backward & forward references as well as inline definition.

Table 3.3 ANS.1 Keywords

Keyword	Brief Description
BEGIN	Start of an ASN.1 module
CHOICE	List of alternatives
DEFINITIONS	Definition of a data type or managed object
END	End of an ASN.1 module
EXPORTS	Data types that can be exported to other modules
IDENTIFIER	A sequence of non-negative numbers
IMPORTS	Data types defined in external modules
INTEGER	Any negative or non-negative number
NULL	A placeholder
OBJECT	Used with IDENTIFIER to uniquely identify an object
OCTET	Unbounded 8-bit bytes (octets) of binary data
OF	Used with SET and SEQUENCE
SEQUENCE	Ordered list maker
SEQUENCE OF	Ordered array of repetitive data
SET	Unordered list maker
SET OF	Unordered list of repetitive data
STRING	Used with OCTET for denoting string of octets

ASN.1 DATA TYPE

Simple Type

- A simple type one for which the values are specified directly. For example, we can define a page of a book as PageNumber of simple type.

i.e. PageNumber::=INTEGER
ChapterNumber::=INTEGER

Structured Type

- A data type is a structured type when it contains other type.
- Types that are within a structured type are called component types. For example ,we can define all the pages of the book as a collection of individual pages.

i.e. BookPages::=SEQUENCE OF
{
SEQUENCE {ChapterNumber , Separator ,PageNumber}
}

- SET is distinguished from SEQUENCE in 2 respects:
  1) The data types should all be distinct and
  2) The order of values in SET is of no consequence whereas it is critical in the SEQUENCE construct.

Tagged Type

- Tagged type is a type derived from another type that is given a new tag id.
- A tagged type is defined to distinguish types within an application.

Other Type

- Other type is a data type that is not predefined.
- This is chosen from CHOICE and ANY types, which are contained in other types.
- Type CHOICE defines the selection of one value from a specified list of distinct types.

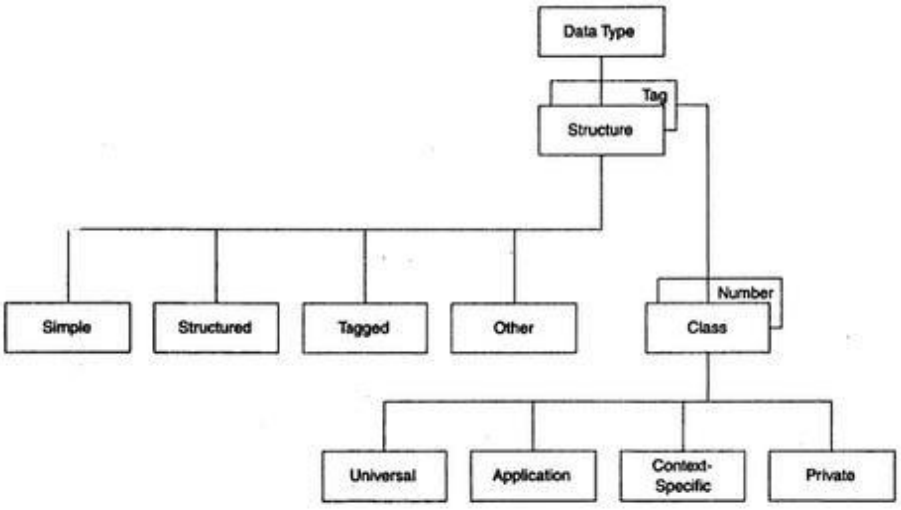

Figure 3.15 ASN.1 Data Type Structure and Tag

ENCODING STRUCTURE

- The ASN.1 syntax that contains the management information is encoded using the BER defined for the transfer syntax.
- The ASCII text data is converted to bit-oriented data.
- Example of encoding structure is TLV which denotes type, length & value components of structure (Fig: 3.18).
- The type has 3 subcomponents: class, P/C & tag number (Table: 3.6).
- P/C specifies whether the structure is a primitive, or simple, type or a construct.
- This is encoded as a one byte (an octet) field.
- The value of P/C is 0 for primitive & 1 for construct.

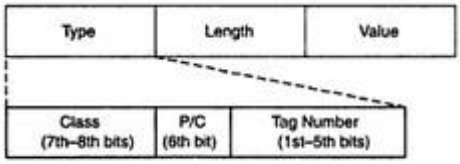

Figure 3.18 TLV Encoding Structure

Table 3.6 Value of Class in Type

Class	8th bit	7th bit
Universal	0	0
Application	0	1
Context-specific	1	0
Private	1	1

FUNCTIONAL MODEL

- The functional model component addresses the user-oriented applications, which are formally specified in the OSI model (Figure: 3.22).
- The functional model consists of 5 submodels: configuration management, fault management, performance management, security management and accounting management.

(same as in chapter 1).

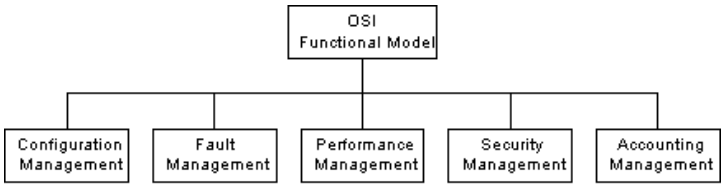

Figure 3.22: functional model