NETWORK MANAGEMENT SYSTEMS

Sub Code	**: 10IS834/10CS834**	**IA Marks**	**: 25**
Hrs/Week	**: 04**	**Exam Hours**	**: 03**
Total Hrs	**: 52**	**Exam Marks**	**: 100**

PART – A

UNIT 1 **7 Hours**

Introduction: Analogy of Telephone Network Management, Data and Telecommunication Network Distributed computing Environments, TCP/IP-Based Networks: The Internet and Intranets, Communications Protocols and Standards- Communication Architectures, Protocol Layers and Services; Case Histories of Networking and Management – The Importance of topology , Filtering Does Not Reduce Load on Node, Some Common Network Problems; Challenges of Information Technology Managers, Network Management: Goals, Organization, and Functions- Goal of Network Management, Network Provisioning, Network Operations and the NOC, Network Installation and Maintenance; Network and System Management, Network Management System platform, Current Status and Future of Network Management.

UNIT 2 **6 Hours**

Basic Foundations: Standards, Models, and Language: Network Management Standards, Network Management Model, Organization Model, Information Model – Management Information Trees, Managed Object Perspectives, Communication Model; ASN.1- Terminology, Symbols, and Conventions, Objects and Data Types, Object Names, An Example of ASN.1 from ISO 8824; Encoding Structure; Macros, Functional Model.

UNIT 3 **6 Hours**

SNMPv1 Network Management - 1 : Managed Network: The History of SNMP Management, Internet Organizations and standards, Internet Documents, The SNMP Model, The Organization Model, System Overview.

UNIT 4 **7 Hours**

SNMPv1 Network Management – 2: The Information Model – Introduction, The Structure of Management Information, Managed Objects, Management Information Base.The SNMP Communication Model – The SNMP Architecture, Administrative Model, SNMP Specifications, SNMP Operations, SNMP MIB Group, Functional Model

PART - B

UNIT 5 **6 Hours**

SNMP Management – RMON: Remote Monitoring, RMON SMI and MIB, RMONI1- RMON1 Textual Conventions, RMON1 Groups and Functions, Relationship Between Control and Data Tables, RMON1 Common and Ethernet Groups, RMON Token Ring Extension Groups, RMON2 – The RMON2 Management Information Base, RMON2 Conformance Specifications; ATM Remote Monitoring, A Case Study of Internet Traffic Using RMON.

UNIT 6 **6 Hours**

Broadband Network Management: ATM Networks: Broadband Networks and Services, ATM Technology – Virtual Path-Virtual Circuit, TM Packet Size, Integrated Service, SONET, ATM LAN Emulation, Virtual LAN; ATM Network Management – The ATM Network Reference Model, The Integrated Local Management Interface, The ATM Management Information Base, The Role of SNMP and ILMI in ATM Management, M1 Interface: Management of ATM Network Element, M2 Interface: Management of Private Networks, M3 Interface: Customer Network Management of Public Networks, M4 Interface: Public Network Management, Management of LAN Emulation, ATM Digital Exchange Interface Management.

UNIT 7 **6 Hours**

Broadband Network Management: Broadband Access Networks and Technologies – Broadband Access Networks, roadband Access Technology; HFCT Technology – The Broadband LAN, The Cable Modem, The Cable Modem Termination System, The HFC Plant, The RF Spectrum for Cable Modem; Data Over Cable Reference Architecture; HFC Management – Cable Modem and CMTS Management, HFC Link Management, RF Spectrum Management, DSL Technology; Asymmetric Digital Subscriber Line Technology – Role of the ADSL Access Network in an Overall Network, ADSL Architecture, ADSL Channeling Schemes, ADSL Encoding Schemes; ADSL Management – ADSL Network Management Elements, ADSL Configuration Management, ADSL Fault Management, ADSL Performance Management, SNMP-Based ADSL Line MIB, MIB Integration with Interfaces Groups in MIB-2, ADSL Configuration Profiles.

UNIT 8 **8 Hours**

Network Management Applications: Configuration Management- Network Provisioning, Inventory Management, Network Topology, Fault Management- Fault Detection, Fault Location and Isolation Techniques, Performance Management – Performance Metrics, Data Monitoring, Problem Isolation, Performance Statistics; Event Correlation Techniques – Rule-Based Reasoning, Model-Based Reasoning, Case-Based Reasoning, Codebook correlation Model, State Transition Graph Model, Finite State Machine Model, Security Management – Policies and Procedures, Security Breaches and the Resources Needed to Prevent Them, Firewalls, Cryptography, Authentication and Authorization, Client/Server Authentication Systems, Messages Transfer Security, Protection of Networks from Virus Attacks, Accounting Management, Report Management, Policy-Based Management, Service Level Management.

Text Books:

1. Mani Subramanian: Network Management- Principles and Practice, Pearson Education, 2003.

TABLE OF CONTENTS

UNIT 1: DATA COMMUNICATION & NETWORK MANAGEMENT OVERVIEW 1-15

UNIT 2: BASIC FOUNDATION — STANDARDS, MODELS & LANGUAGE 16-25

UNIT 3: SNMPV1 NETWORK MANAGEMENT — ORGANIZATION & INFORMATION MODELS 26-28

UNIT 4: SNMPV1 NETWORK MANAGEMENT — ORGANIZATION & INFORMATION MODELS (CONT.) 29-47

UNIT 5: SNMP MANAGEMENT – RMON 48-54

UNIT 1: DATA COMMUNICATION & NETWORK MANAGEMENT OVERVIEW

ANALOGY OF TELEPHONE NETWORK MANAGEMENT

Why Telephone Network is popular?

- This is reliable
- This is dependable
- The Qos is generally good

Telephone Network Model

• A *trunk* is a logical link between two switches that may traverse one or more physical links (Figure: 1.1).

• The customer's telephone which is a switch on the customer premises, is connected to the end office via a dedicated link called a *loop*.

• The direct distance dialing (DDD) network, which enables us to dial far-end telephone w/o an operator's assistance, comprises following 3 transmission trunks:

1) a direct trunk connects 2 end offices
2) a toll connecting trunk connects an end office to any toll office
3) a toll trunk connects any 2 toll offices

• A circuit connection is set up either directly using a local trunk or via the higher level switches & routes.

• Primary & secondary routes are already programmed into the switch. If the primary route is broken or the facilities over the primary route are filled to capacity, an alternative route is automatically assigned.

• Operations support systems ensure the quality of service in the telephone network.

• For a given region, there is a *NOC (Network Operations Center)* where the global status of the network is monitored. The NOC is the nerve center of telephone network operations.

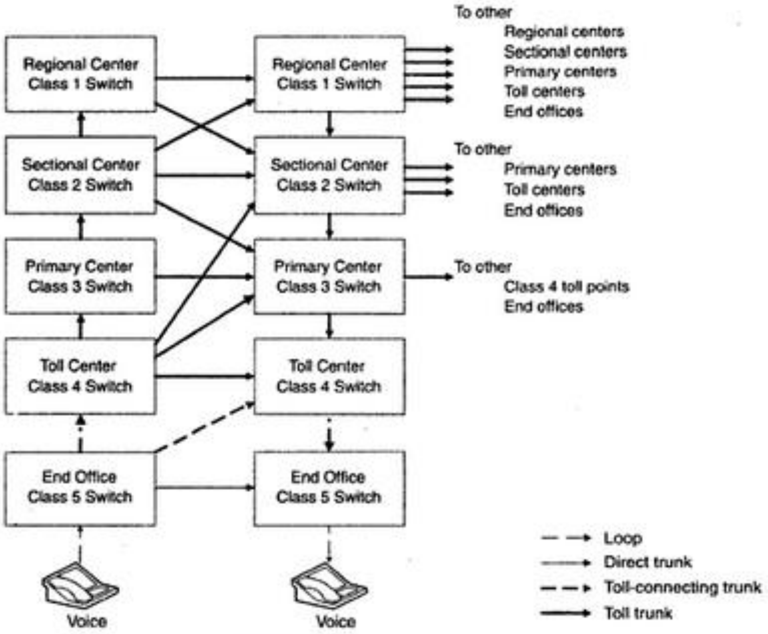

To other
Regional centers
Sectional centers
Primary centers
Toll centers
End offices
Regional Center Class 1 Switch
Regional Center Class 1 Switch
Sectional Center Class 2 Switch
Sectional Center Class 2 Switch
To other
Primary centers
Toll centers
End offices
Primary Center Class 3 Switch
Primary Center Class 3 Switch
To other
Class 4 toll points
End offices
Toll Center Class 4 Switch
Toll Center Class 4 Switch
End Office Class 5 Switch
End Office Class 5 Switch
Voice
Voice
Loop
Direct trunk
Toll-connecting trunk
Toll trunk

Figure 1.1 Telephone Network Model

DATA (COMPUTER) & TELECOMMUNICATION NETWORKS

THREE MODES OF DATA TRANSMISSION

- The data can be transmitted in one of 3 modes:
  1) Circuit switched
  2) Message switched or
  3) Packet switched.
- In the circuit switched mode, a physical circuit is established between the originating & terminating ends before the data is transmitted. The circuit is disconnected after completion of transmission.
- In message-switched & packet-switched modes, the data is broken into packets & each packet is enveloped with the destination & originating addresses.
- Message-switched mode is used to send long messages such as email. Whereas ,Packet switched mode is used to transmit small packets used in applications such as interactive communication.
- In message switched mode, the data is stored by the system & then retrieved by the user at a later time. In packet switched mode, the packets are fragmented & reassembled in almost real time.
- The bridges & routers open each packet to find the destination address & switch the data to the appropriate output links.

DATA & TELECOMMUNICATION NETWORKS

- Telecommunication network is a circuit-switched network that is structured as a public network accessible by any user (Figure: 1.3).
- The organization that provides service is called a telecommunication service provider E.g. BSNL, Airtel.
- To interface, a terminal or host connected to an end-office switch communicates with the host connected to another end-office switch by modems at each end.
- Modems transfer the information from digital to analog at source & back to digital at destination.

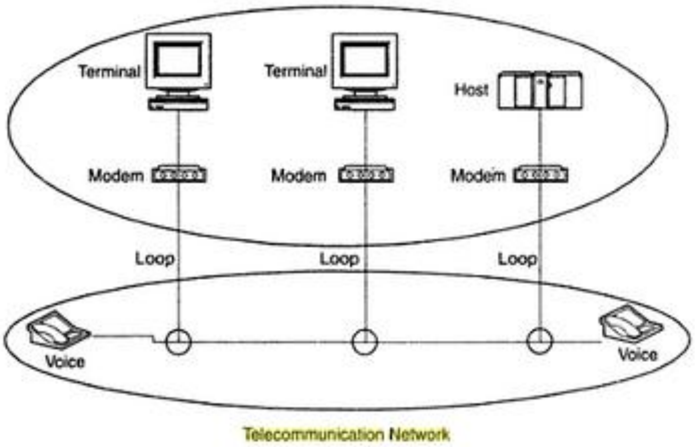


Figure 1.3 Data and Telecommunication Networks

INTERIM CORPORATE DATA & TELECOMMUNICATION NETWORK

- A number of telephones & computers terminals at various corporate sites are connected by the telecommunication network (Figure: 1.4).
- The telephone are connected locally by a local switch, PBX, which interfaces to the telephone network.
- The computer terminals are connected to onsite communication controllers, which manages the local terminals & provides a single interface to the telephone network.
- In the above corporate environment, the computer terminals communicate directly with the host.
- This communication system architecture is expensive & inefficient because the user has to pay for the data traffic over the public or leased telecommunications line.
- To reduce the cost & improve the performance, the computer terminals can communicate with a local communications processor, which can then communicate with remote hosts.
- Processor-to-processor communications over the telecommunications lines takes less time & therefore are less expensive.

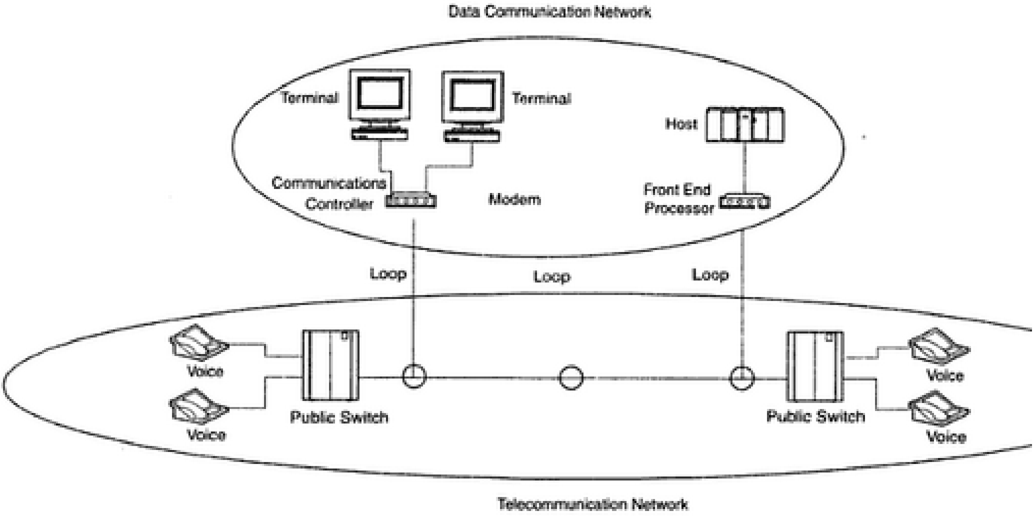


Figure 1.4 Interim Corporate Data and Telecommunication Networks

IBM SYSTEMS NETWORK ARCHITECTURE MODEL

- In SNA, the host is connected to the terminals via the communications controllers & cluster controllers.
- Cluster controllers manage the DTEs at the peripheral nodes & the communication controllers manage the traffic at the subnetwork levels (Figure: 1.5).

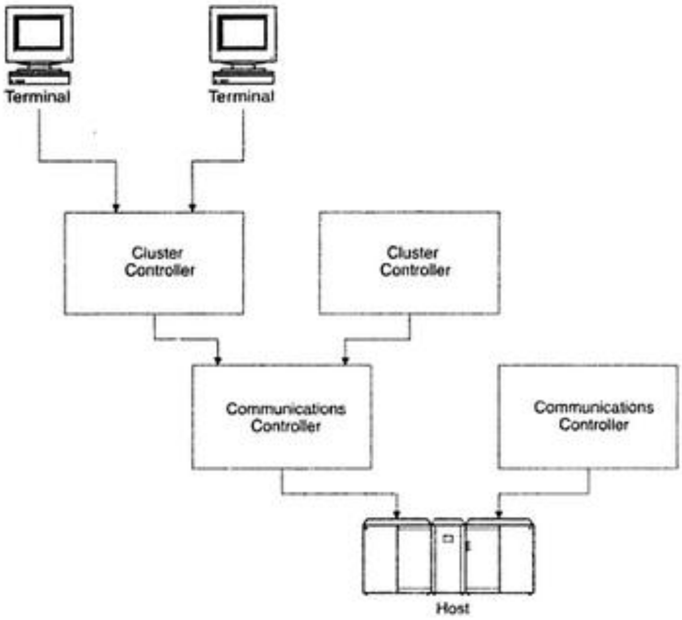


Figure 1.5 IBM Systems Network Architecture Model

DCE (DISTRIBUTED COMPUTING ENVIRONMENT)

SIMPLE CLIENT/SERVER MODEL

- The client initiates a request to the server & waits (Figure: 1.7).
- The server executes the process to provide the requested service & sends the results to the client.
- The client cannot initiate a process in the server. Thus, the process should have already been started in the server & be waiting for requests to be processed.

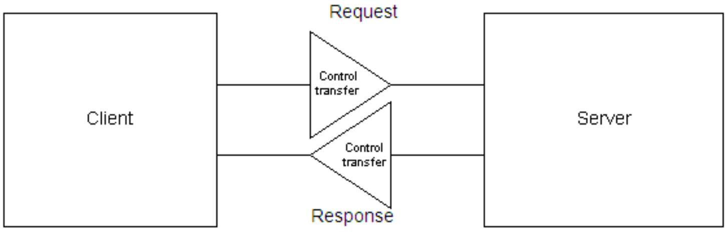


Figure 1.7 Simple Client-Server Model

MODEL OF CLIENT/SERVER NETWORK IN A DCE

- Each client's request is normally processed by the server according to the FIFO rule (Figure: 1.8,). This delay could be minimized, but not eliminated by concurrent processing of requests by the server.
- Since the client & application processes are running in a distributed computing environment, each of them can be designed to execute a specific function efficiently.
- For example, joe.stone using a client in a network sends a message to sally.jones@dest.com on the network.
- The message first goes to the mail server on the network. Before it can process the request, the mail server needs to know the network address of sally.jones, which is dept.com.. Therefore, it makes a request to the DNS on the network for the routing information for the address of dept.com
- When it receives that information, it sends out joe.stone's message via the bridge to the network.
- In this example, the mail server behaves both as a server & as a client.
- The 3 processes in this scenario, namely the client, the mail server and the DNS are considered cooperative computing processes & may be running in 3 separate platforms on remote LANs connected by a WAN. The communication between these processes is called peer-to-peer communication.

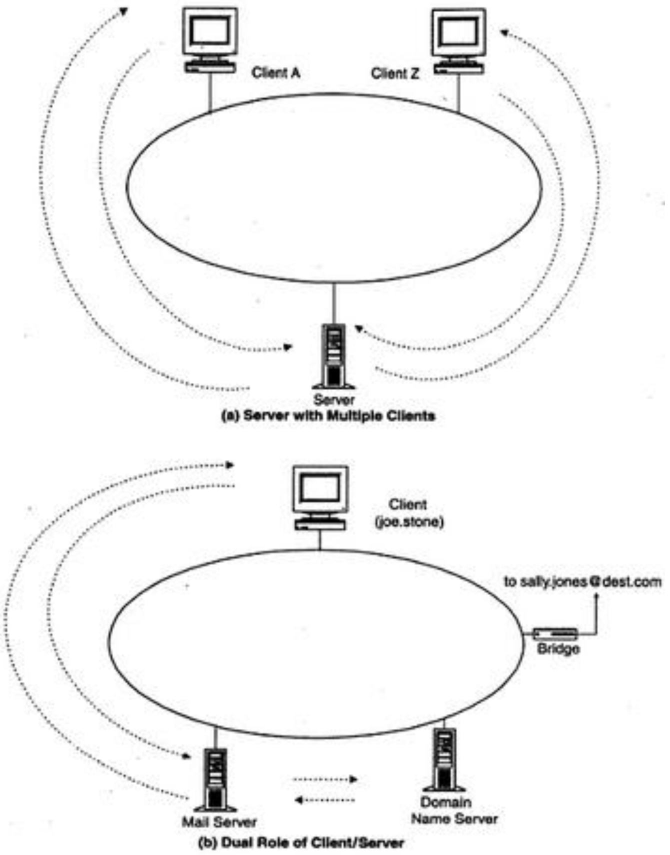


Figure 1.8 A Model of a Client/Server Network in a Distributed Computing Environment

TCP/IP-BASED NETWORKS: THE INTERNET AND INTRANET

- TCP/IP is a suite of protocols that enable networks to be interconnected.
- TCP/IP forms the basic foundation of the Internet(Figure:1.9).
- The nodes in the network use network protocol named IP to route packets.
- IP is a connectionless protocol. That means there is no guarantee that the packets will be delivered to the destination node. However, end-to-end communication can be guaranteed by using the transport protocol, TCP.
- TCP is connection-oriented protocol. Whereas , UDP is a connectionless protocol.
- Much of Internet traffic really uses UDP/IP, because of the reliability of data transmission.
- The Internet is a network of networks. Whereas, An intranet is a private network & access to it is controlled by the enterprise that owns it, whereas the Internet is public.
- Gateways between LANs serve as the interfaces between dissimilar & independent, autonomous networks & perform many functions including protocol conversions.

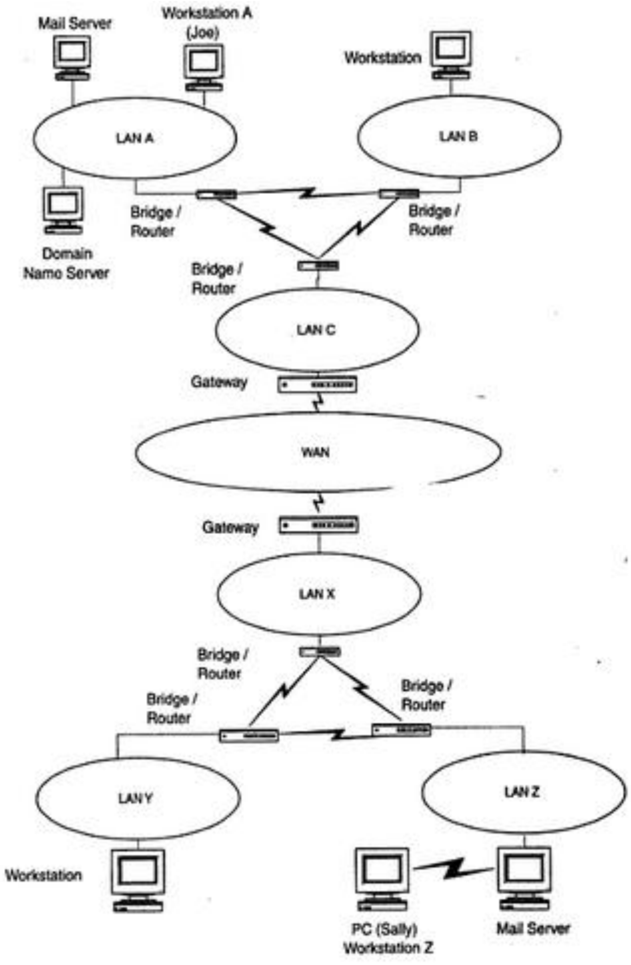


Figure 1.9 Internet Configuration

INTERNET FABRIC MODEL

- The workstations belong to the user plane, the LANs to the LAN plane, & WANs to the WAN plane.
- The interfaces are defined as the fabrics (Figure: 1.10).
- MAC fabric interfaces the user plane & the LAN plane. The user's workstation interfaces to a LAN via a MAC
- LANs interface to a WAN by a switching fabrics of bridges, routers & switches.
- Each WAN can be considered an autonomous network, & hence needs a gateway to communicate with another WAN. Gateway fabric interconnects different WANs.

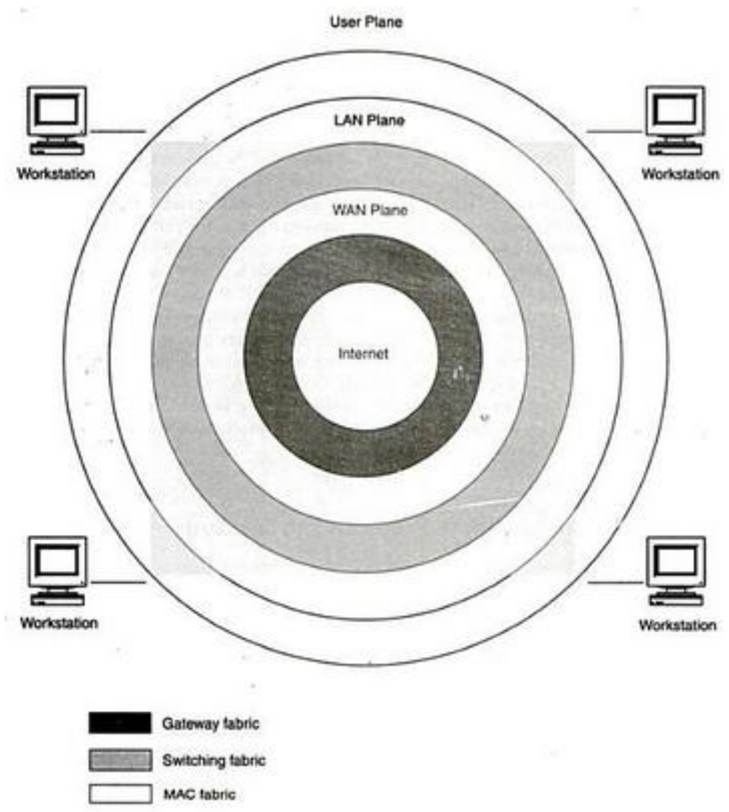
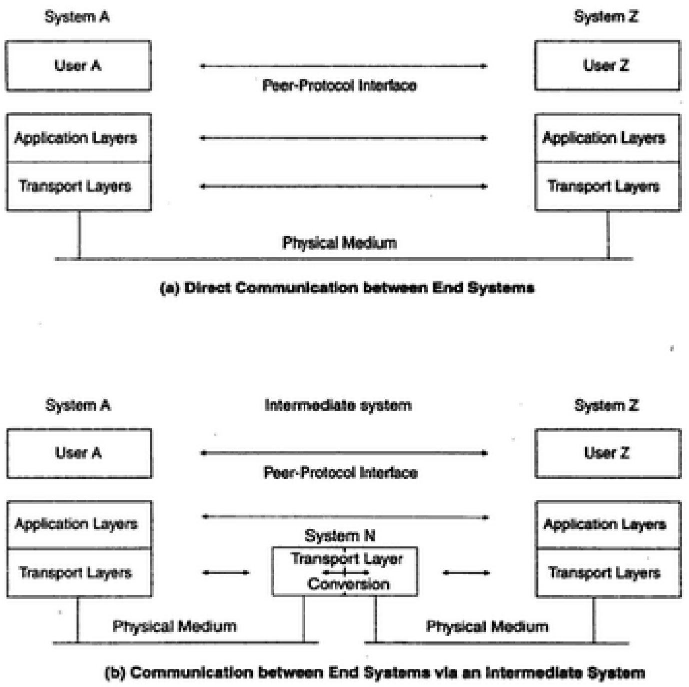


Figure 1.10 Internet Fabric Model

COMMUNICATION PROTOCOLS AND STANDARDS

COMMUNICATION ARCHITECTURE

- Communication between users occurs at various levels.
- Each system can be divided into 2 broad sets of communication layers. The top set of layers consists of the application layers & the bottom set of the transport layers.
- The users interface with the application level layer & the communication equipment interfaces with the physical medium.
- In Figure:1.11a, direct communication occurs between the corresponding cooperating layers of each system.
- In Figure:1.11b, the end systems communicating via an intermediate system N, which enables the use of different physical media for the 2 end systems.
- System N converts the transport layer information into the appropriate protocols.



(a) Direct Communication between End Systems

(b) Communication between End Systems via an Intermediate System

Figure 1.11 Basic Communication Architecture

OSI COMMUNICATION ARCHITECTURE

- OSI model was developed based on the premise that the different layers of protocol provide different services, and that each layer can communicate with only its own neighboring level (Figure: 1.12).
- Two systems can communicate on a peer-to-peer level i.e. at the same level of the protocol.

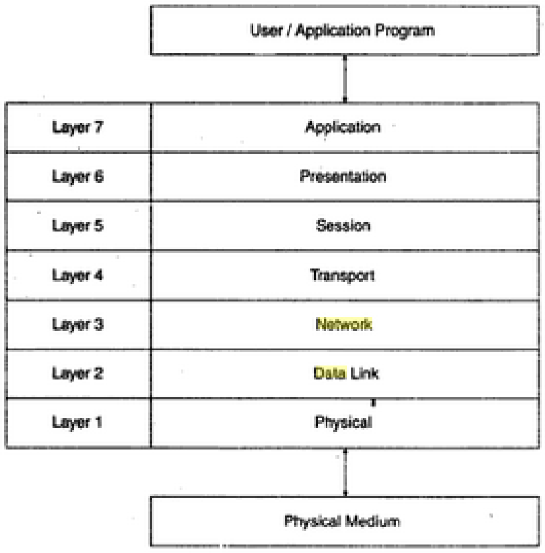


Figure 1.12 The OSI Protocol Layers

PDU COMMUNICATION MODEL BETWEEN END SYSTEMS

- The message in each layer is contained in message units called protocol data units (PDUs), which consists of two parts-- PCI(protocol control information) & UD(user data) (Figure:1.14).
- PCI contains header information about the layer.
- UD contains the data that the layer, acting as a service provider, receives from or transmits to the upper layer/service user layer.
- The size of the PDU increases as it goes toward lower layers.
- If the size of the PDU exceeds the maximum size of layers specifications, it is fragmented into multiple packets. Thus, a single application-layer PDU could multiply into several physical PDUs.

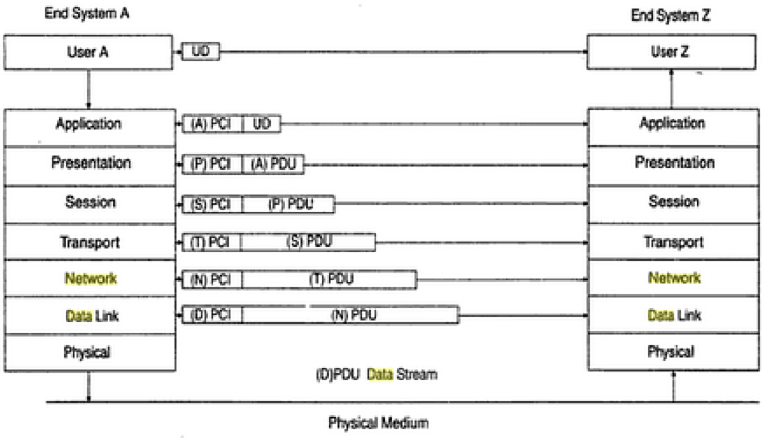


Figure 1.14 PDU Communication Model between End Systems

OSI LAYERS & SERVICES

Physical layer

- Transfers to & gathers from the physical medium raw bit data (Figure: 1.13).
- Handles physical & electrical interfaces to the transmission medium.

Data link layer

- Consists of two sublayers: LLC(Logical link control) & MAC(Medium access control).
- LLC formats the data to go on the medium, performs error control & flow control.
- MAC controls data transfer to & from LAN, resolves conflicts with other data on LAN.

Network layer

- Forms the switching/routing layer of the network.

Transport layer

- Multiplexes & demultiplexes messages from applications.
- Acts as a transparent layer to applications & thus isolates them from the transport system layers.
- Makes & breaks connections for connection-oriented communications.
- Controls flow of data in both directions.

Session layer

- Establishes & clears sessions for applications, and thus minimizes loss of data during large data exchange.

Presentation layer

- Provides a set of standard protocols so that the display would be transparent to syntax of the application.
- Data encryption & decryption.

Application layer

- Provides application-specific protocols for each application & each transport protocol system.

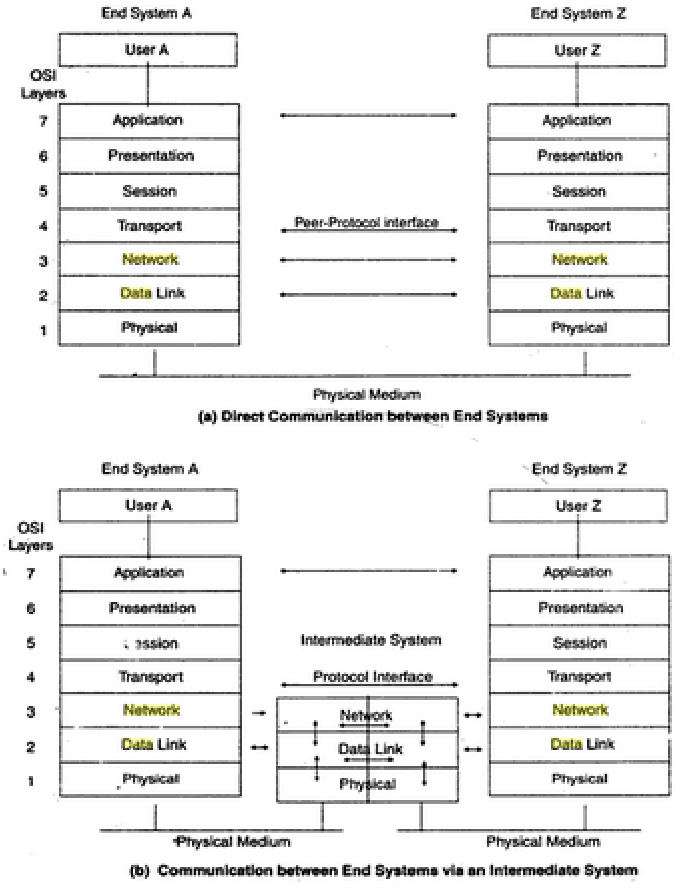


Figure 1.13 OSI Communication Architecture

PHYSICAL LAYER

- This is responsible for physically placing the electrical signal on the physical medium & picking up the signal from it.
- This controls & manages the physical & electrical interfaces to the physical medium, including the connector or transceiver.
- The physical medium could be copper, optical fiber or wireless media.
- The signal could be either digital or analog.

DATA LINK LAYER

- The data communication between 2 DTEs is controlled & managed by this layer.
- The data communication is a serial bit-oriented stream.
- The data link layer needs to do basic functions:
  1) Establish & clear the link, and
  2) Transmit the data.
- This does error control & data compression. Flow control is done on a hop-to-hop basis.
- This is divided into two sublayers--LLC & MAC (Figure: 1.15). The lower MAC layer controls the access & transmittal of data to the physical layer in an algorithmic manner. LLC performs the link management & data transfer.
- There are two basic forms of LANs--Ethernet LAN is a bus type & the media is accessed using a distributed probabilistic algorithm, CSMA/CD The second type of LAN is a ring type used in token ring & FDDI.A deterministic token-passing algorithm is used in this case.

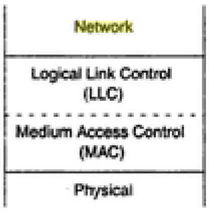


Figure 1.15 The Sublayer Structure of a Data Link Protocol Layer

NETWORK LAYER

- This controls & manages the switching fabric of the network (Figure: 1.16).
- This provides both CLNS (connectionless network services) & CONS (Connection oriented network service). CLNS is used when the lower layers are highly reliable such as LANs & bridges as well as when the messages are short. CONS is the method for transmitting long messages such as file transfer. This is also used when the transmission medium is not reliable.
- The OSI architecture model handles this by dividing the network layer into 3 sublayers:
  1) SNICP (Subnetwork Independent Convergence Protocol)
  2) SNDCP (Subnetwork Dependent Convergence Protocol)
  3) SNDAP (Subnetwork Dependent Access Protocol) (Figure: 1.17)
- The Internet communicates between nodes using a Internet address and the SNICP. The nodes in turn communicate with subnetworks using the SNDCP, which depends on the subnetwork protocol & could be any proprietary protocol. In such a situation, the SNDCP communicates with its data link layer via the SNDAP.

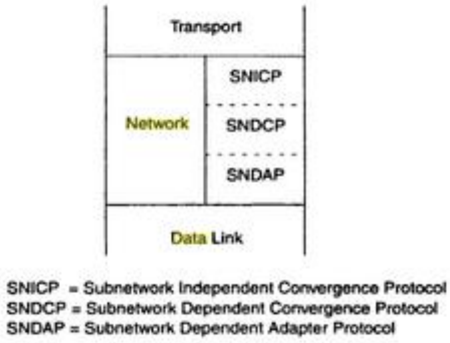
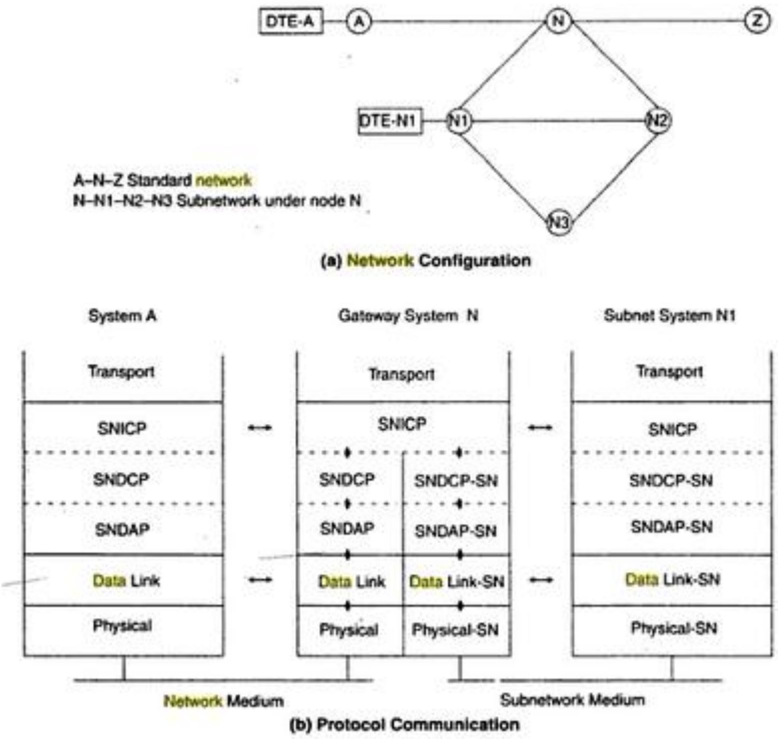

Transport
SNICP
Network
SNDCP
SNDAP
Data Link
SNICP = Subnetwork Independent Convergence Protocol
SNDCP = Subnetwork Dependent Convergence Protocol
SNDAP = Subnetwork Dependent Adapter Protocol

Figure 1.16 The Sublayer Structure of a Network Protocol Layer



(a) Network Configuration

(b) Protocol Communication

Figure 1.17 Gateway Communication to a Proprietary Subnetwork

PRESENTATION LAYER

- This is the medium of presentation of the message's context to the user or application program.
- This is context sensitive layer.
- This can be interpreted as the common languages & image that the users at both end systems use & understand.

COMPARISON OF SNA, OSI AND INTERNET PROTOCOL LAYER MODELS

- The transport & network layers form the suite of TCP/IP protocols. The application layers are combined into application-specific protocols (Figure: 1.18).
- In the 7-layered SNA model, the physical, data link & application layers have one-to-one correspondence with the OSI layers.
- Much of the SNA transport & session layer functions equivalent to those of the OSI model are done in the data flow control & transmission control layers. The combination of these 2 services is also called *the SNA transmission subsystem.*
- The presentation services, which are known as SNA high level services, combine the presentation services & functions with some of the session control functions.

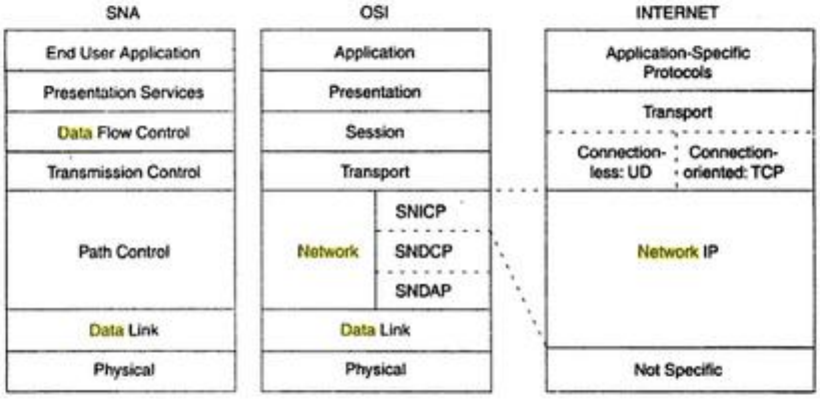


Figure 1.18 Comparison of SNA, OSI, and Internet Protocol Layer Models

APPLICATION SPECIFIC PROTOCOLS IN THE ISO & INTERNET MODELS

- All application specific protocol services in OSI are sandwiched between the user & presentation layers. In the Internet model, they are sandwiched between the user and the transport layers (Figure: 1.19).
- A user interfaces with a host at a remote terminal using virtual terminal in the OSI model & TELNET in the Internet model.
- File transfers are accomplished using FTAM (File Transfer Access & Management) in the OSI model and FTP (File transfer protocol) in the Internet.
- The most common mail service function in the Internet is the SMTP.A similar protocol in the OSI model is the MOTIS (message oriented text interchange standard).
- Network management is accomplished using CMIP (Common Management Information protocol) in the OSI model and SNMP in the Internet.

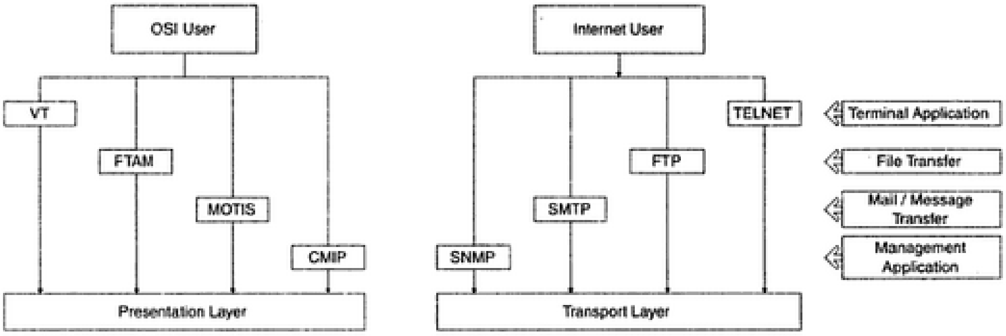


Figure 1.19 Application-Specific Protocols in the ISO and Internet Models

CHALLENGES OF INFORMATION TECHNOLOGY MANAGERS

- IT manager needs to maintain both computer & telecommunication networks because both types are slowly merging in function.
- They are responsible for management of information because of the explosion of information storage & transfer in the modern information era.
- They have to keep up with the new technologies because the technology is moving fast & the corporate growth is enormous.
- They need to make provisions for contingencies to change direction when the IT industry does
- They face network & administrative & management problems day in & day out because most of the corporate networks run 24/7.

WHAT ARE TOP CHALLENGES IN MANAGING THE NETWORK?

- Analyzing problems, which requires intuition & skill
- Anticipating customers' demands
- Acquiring resources
- Managing the client/server environment
- Networking with emerging technology as part of continuing education
- Collaborative research between academic institutions & industry
- Maintaining reliability
- Diagnosing problems or outrages in a non-disruptive manner
- Estimating value of a technology transition
- Maintaining a secure firewall between the internal network & the Internet
- Sustainable network that is scalable & maintainable
- Staying abreast of the rapid advance of technology
- Determining responsibility for outages to the WAN

NETWORK MANAGEMENT: GOALS, ORGANIZATIONS & FUNCTIONS

- This can be defined as OAM&P of network & services.
- The goal of network management is to ensure that the users of a network receives the information technology services with the quality of service that they expect.

Network Management Functions

Network Provisioning

- The engineering group keep track of new technologies & introduces them as needed. (Figure: 1.21).
- Determination of what is needed & when is made through analysis of the traffic and performance data provided by the network operations.
- Network management tools are helpful to the engineering group in gathering statistics and studying the trends of traffic patterns for planning purposes.

Network Operations & the NOC

- They are concerned with daily operations of the network & providing network services.

Fault Management/Service Restoration

(Check detailed FM in next page).

Trouble Ticket Administration

- This is the administrative part of fault management & is used to track problems in the network. All problems, including nonproblems, are to be tracked until resolved.

Configuration Management

- There are 3 configurations of the network:
  1) One is the static configuration & is the permanent configuration of the network. The static configuration is on that would come up if the network is started from idle status.
  2) The second configuration of a network is the current running configuration.
  3) The third configuration is the planned configuration of the future when the configuration data will change as the network is changed. This information is useful for planning & inventory management.

Security management

(Check detailed SM in next page).

Performance Management

(Check detailed PM in next page).

Accounting Management

- The NOC administers costs & allocates the use of the network.
- Metrics are established to measure the usage of resources & services.
- There are 3 classes of reports: systems, management & user.

Network Installation & Maintenance

- The network I&M group takes care of all installation & maintenance of equipment & cables.
- This group is the service arm of the engineering group for installation & fixing troubles for network operations.

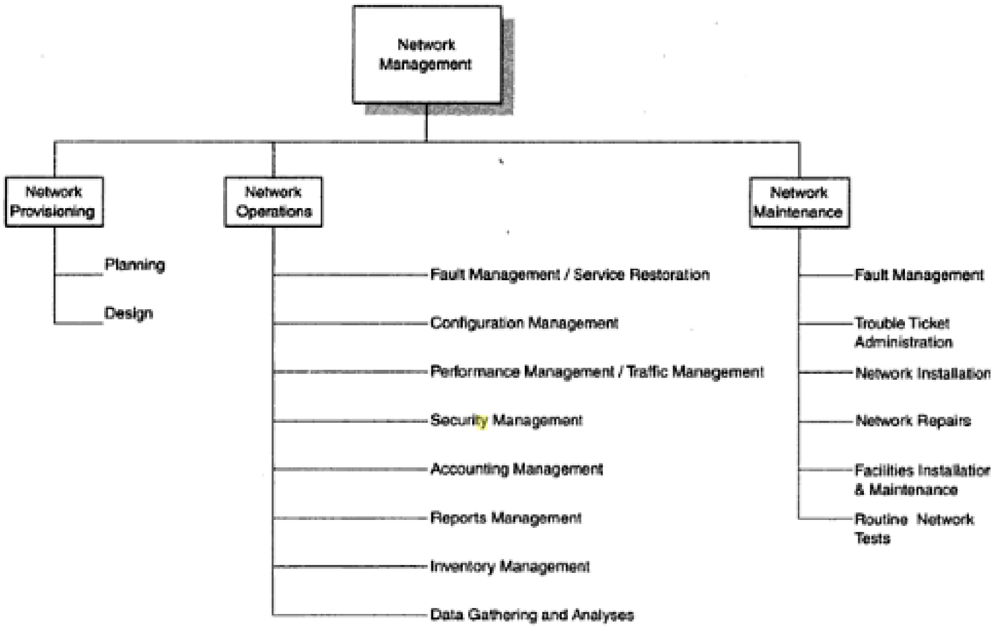


Figure 1.21 Network Management Functional Groupings

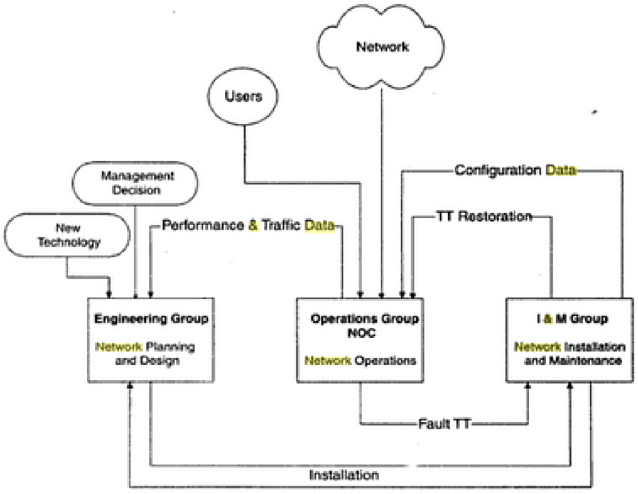


Figure 1.22 Network Management Functional Flowchart

NETWORK OPERATIONS & THE NOC (in detail)

FAULT MANAGEMENT

- This involves detection & isolation of the problem that caused the failures, and restoration of the service.
- Whenever there is a service failure it is NOC's responsibility to restore service as soon as possible. In several failure situations, the network will do this automatically. This network feature is called *self-healing*.
- An NMS can also detect failures of components & indicate them with appropriate alarms.
- The responsibility to fix the problem usually rests with the I&M group.
- A trouble ticket is generated manually by a source engineer at NOC using a trouble-ticket system or automatically generated by an NMS.
- The information on the trouble ticket includes

→ a tracking number assigned by the system → time at which problem occurred
→ the nature of the problem → affected user
→ the responsible group/engineer to resolve the problem

- The tracking of a trouble involves several groups and the administration of it generally belongs to the network maintenance group.

SECURITY MANAGEMENT

- This involves physically securing network, access to network resources & secured communication over network.
- Access privilege to application software is not the responsibility of the NOC unless the application is either owned or maintained by the NOC.
- A security database is established & maintained by the NOC for access to the network & network information.
- Unauthorized access to the network generates an alarm on the NMS at the NOC.
- Firewalls protect corporate networks & network resources from being accessed by unauthorized personnel & programs including virus programs.
- Secured communication prevents tampering of information as it traverses the network, so that is cannot be accessed or altered by unauthorized personnel. Cryptography plays a vital part in security management.

PERFORMANCE MANAGEMENT

- This is concerned with the performance behavior of the network.
- The status of the network is displayed by a NMS that measures the traffic & performance of the network.
- The NOC gathers data & keeps them up to date to tune the network for optimum performance.
- The network statistics include data on traffic, network availability& network delay.
- The traffic data can be captured based on volume of traffic in the various segments of the network.
- Performance data on availability & delay is useful for tuning the network to increase the reliability & to improve its response time.
- Traffic statistics are helpful in detecting trends & planning future needs.

ACCOUNTING MANAGEMENT

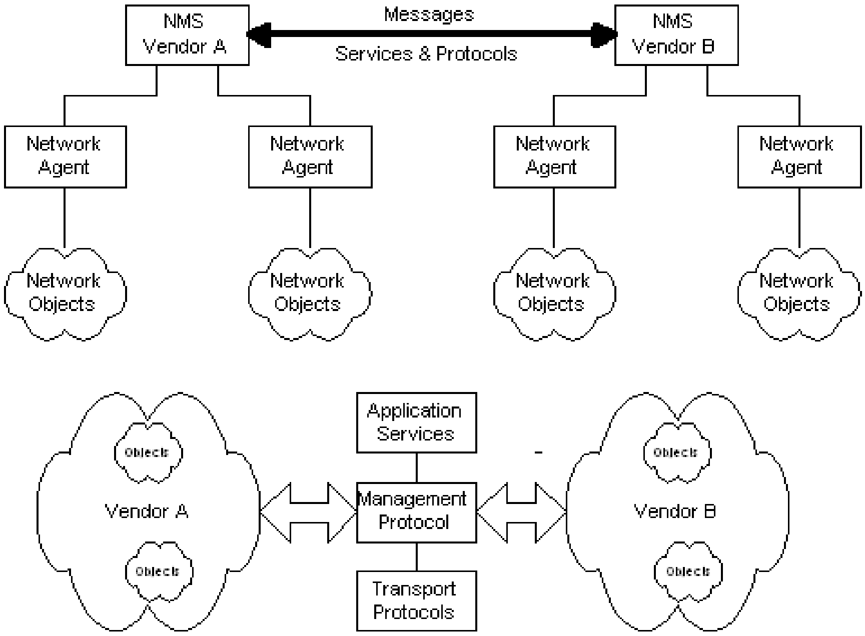
- The NOC administers costs & allocates the use of the network.
- Metrics are established to measure the usage of resources & services.
- There are 3 classes of reports: systems, management & user.
- System reports are needed for network operations to track the activities. Management reports go to the management of the network management group to keep them informed about the activities & performance of the NOC & the network. The user reports are distributed to the users on a periodic basis to let them know the status of network performance.

NETWORK & SYSTEM MANAGEMENT

- The problem in the application program is a system problem & falls under the category of system management. On the other hand, the transport problem from the client's workstation to the server platform is a system problem & falls under network management.
- System management is concerned with management of systems & system resources in the network. Whereas, Network management is concerned with network resources such as hubs, switches, bridges, routers & gateways, and the connectivity among them via a network.
- Network management also addresses end-to-end connectivity between any two processors in the network. System management also addresses logging & archiving events.

Network Management Dumbbell Architecture

- In fig:1.23 , the messages consist of management information data & management controls.
- Application services are the management-related applications such as fault & configuration management.
- The management protocols are CMIP for the OSI model & SNMP for the Internet model.
- Transport protocols are first 4 layers of OSI model & TCP/IP over any of first 2 layers of the 7-layer OSI model.



(b) Services and Protocols

Figure 1.23 Network Management Dumbbell Architecture

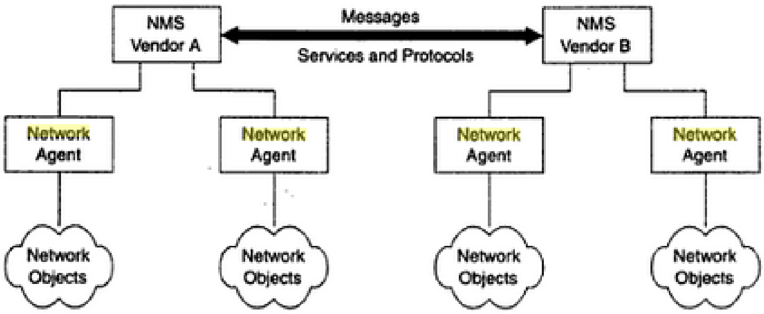


Figure 1.25 Network Management Interoperability

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.

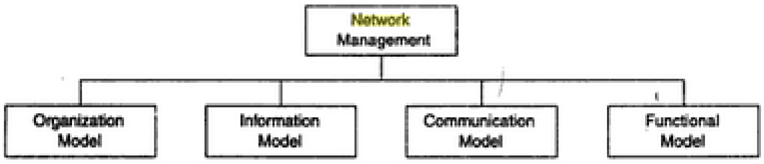


Figure 3.1 OSI Network Management Model

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 (Figure: 3.4).

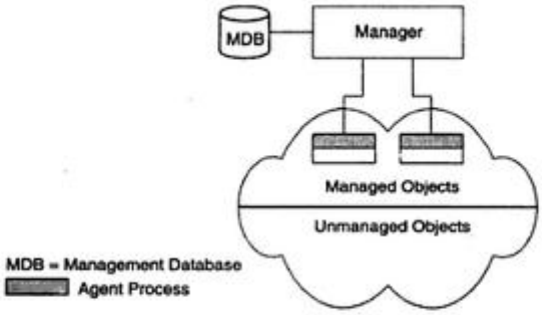


Figure 3.2 Two-Tier Network Management Organization Model

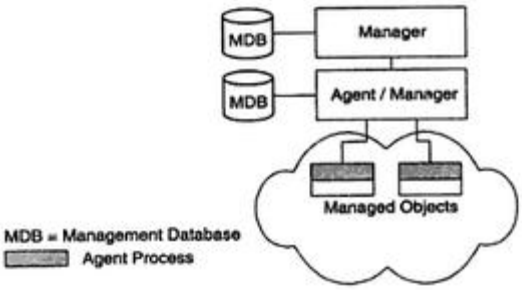


Figure 3.3 Three-Tier Network Management Organization Model

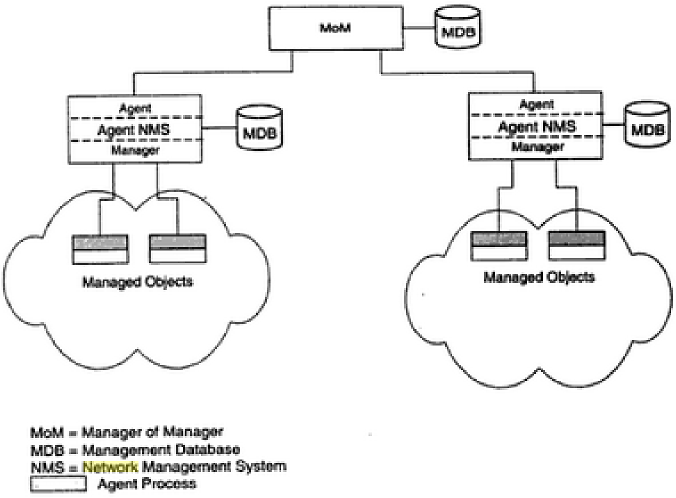


Figure 3.4 Network Management Organization Model with MoM

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.

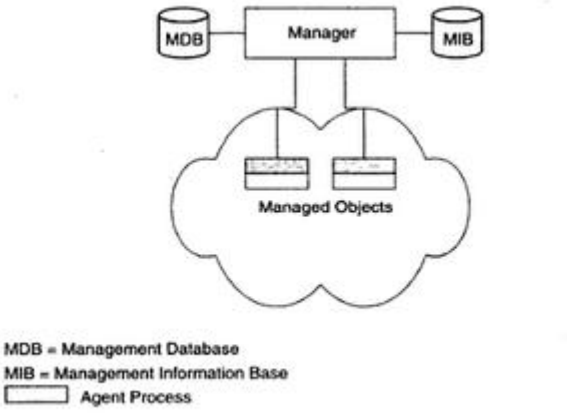


Figure 3.6 Network Configuration with Data and Information Base

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.

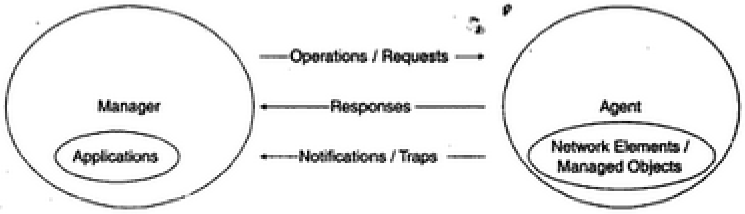


Figure 3.11 Management Communication Model

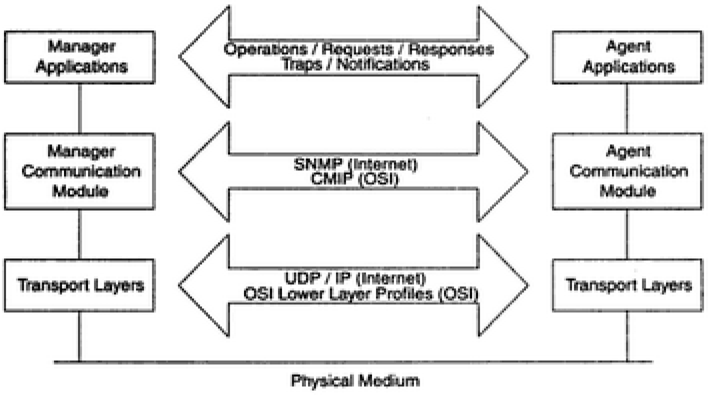

Manager Applications
Operations / Requests / Responses Traps / Notifications
Agent Applications
Manager Communication Module
SNMP (Internet) CMIP (OSI)
Agent Communication Module
Transport Layers
UDP / IP (Internet) OSI Lower Layer Profiles (OSI)
Transport Layers
Physical Medium

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.

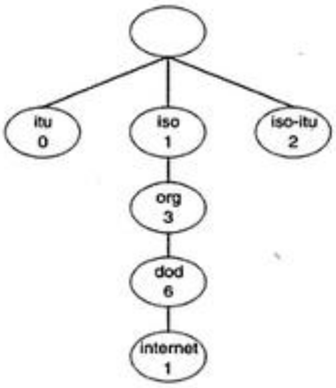


Figure 3.8 OSI Management Information Tree

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.

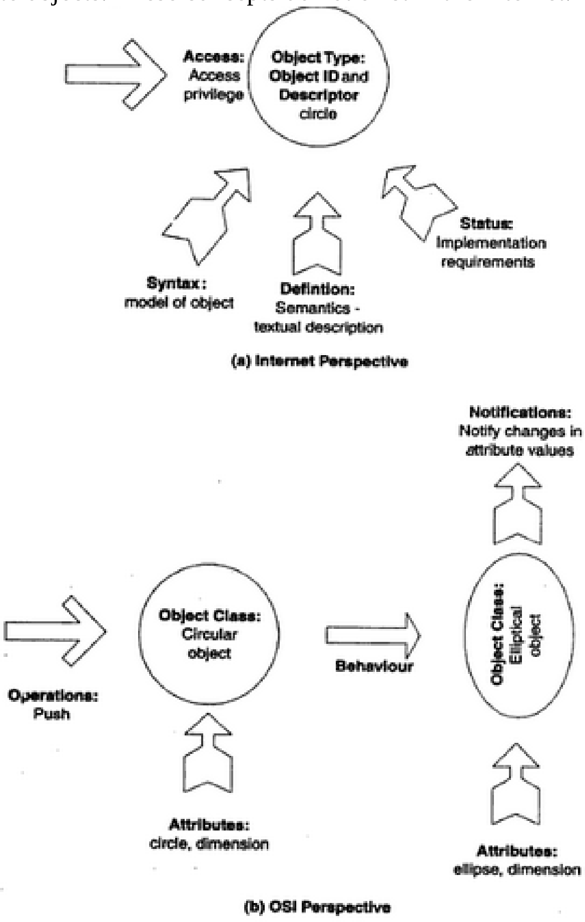
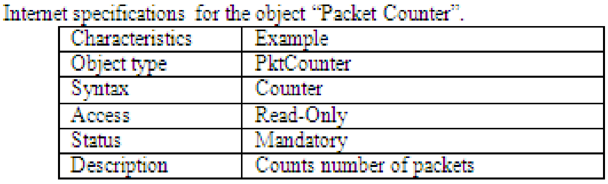
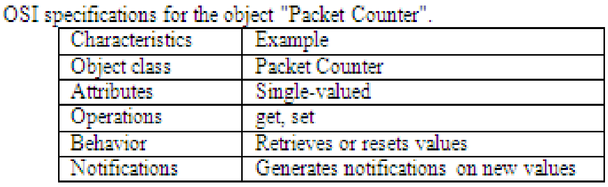

Access: Access privilege
Object Type: Object ID and Descriptor circle
Status: Implementation requirements
Syntax: model of object
Defintion: Semantics - textual description
(a) Internet Perspective
Notifications: Notify changes in attribute values
Object Class: Circular object
Behaviour
Object Class: Elliptical object
Operations: Push
Attributes: circle, dimension
Attributes: ellipse, dimension
(b) OSI Perspective

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.

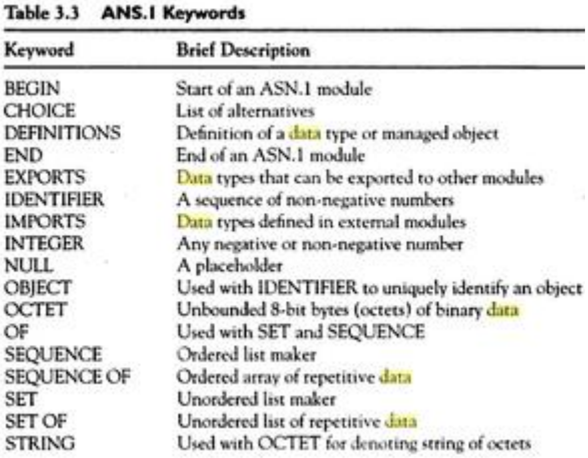


Table 3.3 ANS.1 Keywords

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.

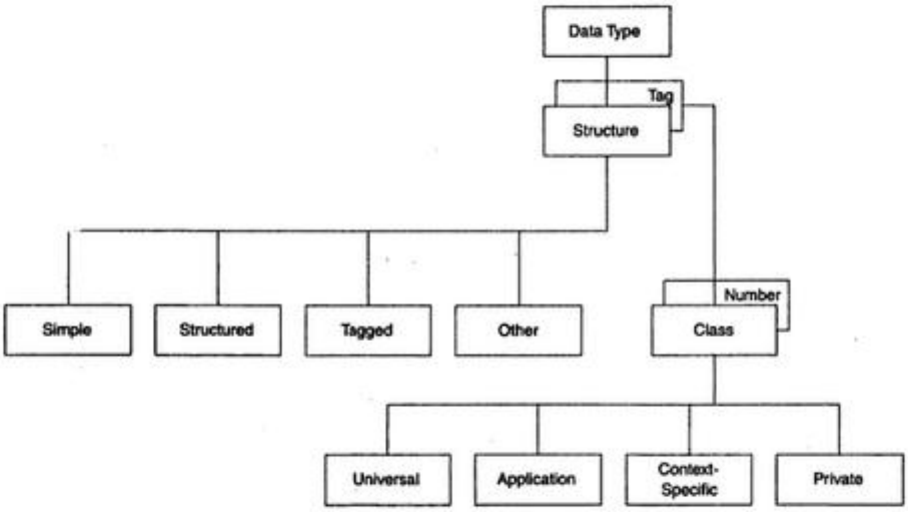


Figure 3.15 ASN.1 Data Type Structure and Tag

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.

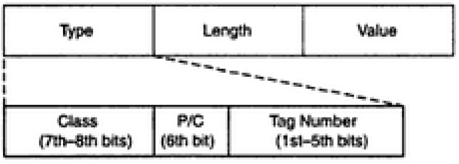
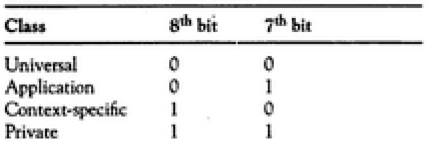


Figure 3.18 TLV Encoding Structure

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).

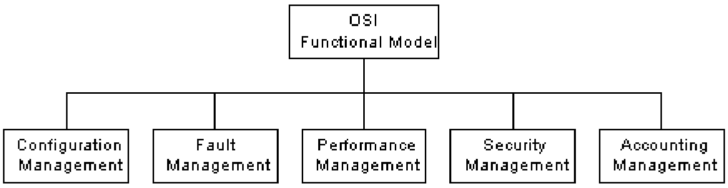


Figure 3.22: functional model

UNIT 3: SNMPv1 NETWORK MANAGEMENT—ORGANIZATION & INFORMATION MODELS

MANAGED LAN NETWORK

- An NMS can automatically discover any component in the network as long as the component has a management agent. For eg, the management agent may be TCP/IP suite that responds to a ping by the NMS (fig:4.1).
- Ethernet LAN consists of a router & 2 hubs and is connected to the backbone network .
- The IP address 172.16.46.1 is the address assigned to the interface card in the router.
- The interface cards in the router and the interface card in each of the hubs are connected by a cat-5 cable, forming the Ethernet LAN.
- The NMS (whose IP address is 192.168.252.1) is physically & logically located remotely from the 172.16.46.1 LAN.
- Information system managers establish conventions to designate a network and a subnetwork. A 0 in the 4th decimal position of an IP address designates a network and 1 in the 4th decimal position designates a subnetwork.
- Once the network components have been discovered & mapped by the NMS ,we can query & acquire information on system parameters and statistics onthe network elements.

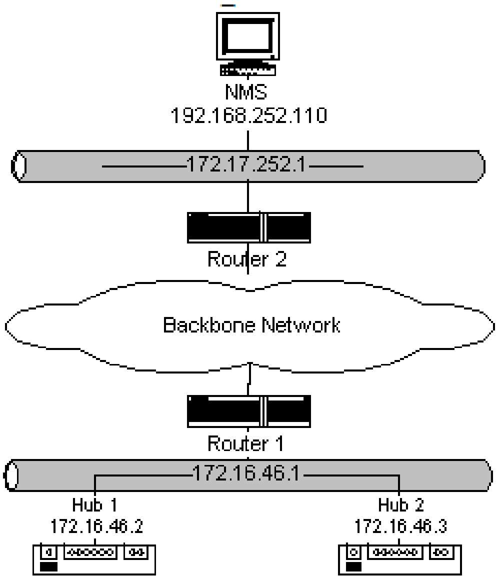


Figure 4.1 A Managed LAN Network

SNMP MODEL

- Organization Model
  - → Relationship between network element,
  - → Agent, and manager
  - → Hierarchical architecture
- Information Model
  - → Uses ASN.1 syntax
  - → SMI (Structure of Management Information
  - → MIB (Management Information Base)
- Communication Model
  - → Transfer syntax
  - → SNMP over TCP/IP
  - → Communication services addressed by messages
  - → Security framework community-based model

SNMP ORGANIZATION MODEL

Two-Tier Model

- This consists of an agent process, which resides in the managed object, and a manager process, which resides in the NMS and manages the managed object. (Fig:4.5).
- Both the manager and the agent are software modules.
- The agent responds to any NMS that communicates with it using SNMP. Thus, multiple managers can interact with one agent.
- In the 2-tier models, the network manager receives raw data from agents & processes them. Sometimes, it is beneficial for the network manager to obtain preprocessed data. Instead of the network manager continuously monitoring the events and calculating the information, an intermediate agent called RMON is inserted between the managed object and the network manager.

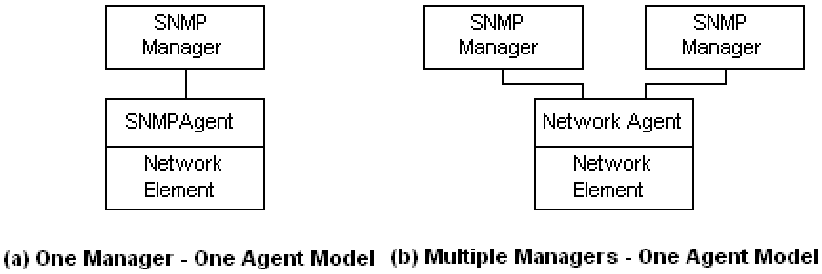
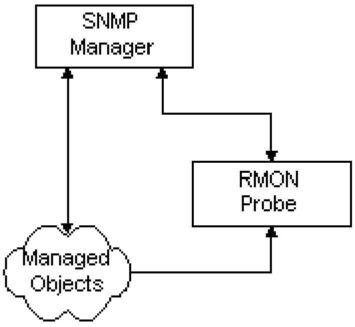


Figure 4.5: two tier organization model

Three-Tier Model

- In **3-tier organization model**, the network manager receives data from the managed objects as well as data from the RMON agent about the managed objects (Fig: 4.6).
- The RMON function has greatly increased the centralized management of networks.



Figyre 4.6: Three tier organization model

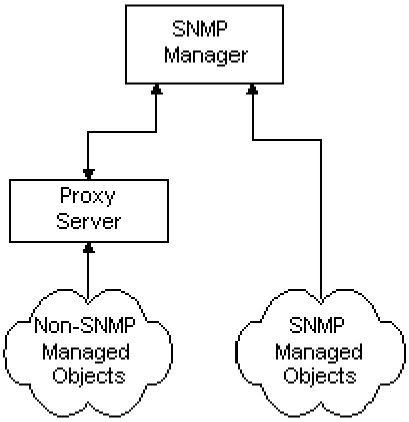


Figure 4.7: Three tier organization model with proxy server

Three-Tier Model with Proxy Server

- Normally, the pure SNMP management system consists of SNMP agents and SNMP managers. However, an SNMP manager can manage a network element that does not have an SNMP agent. This is shown in fig: 4.7
- This model is applicable in many situations, such as legacy systems management, telecommunications network management, wireless networks management and so on.
- A proxy server converts the data into a set that is compatible with SNMP and communicates with the SNMP manager.

SNMP NETWORK MANAGEMENT ARCHITECTURE

- This portrays the data path between the manager application process and the agent application process via the 4 transport protocols: UDP, IP, DLC & PHY. The 3 application layers above the transport layer are integrated in the SNMP process (fig: 4.9).

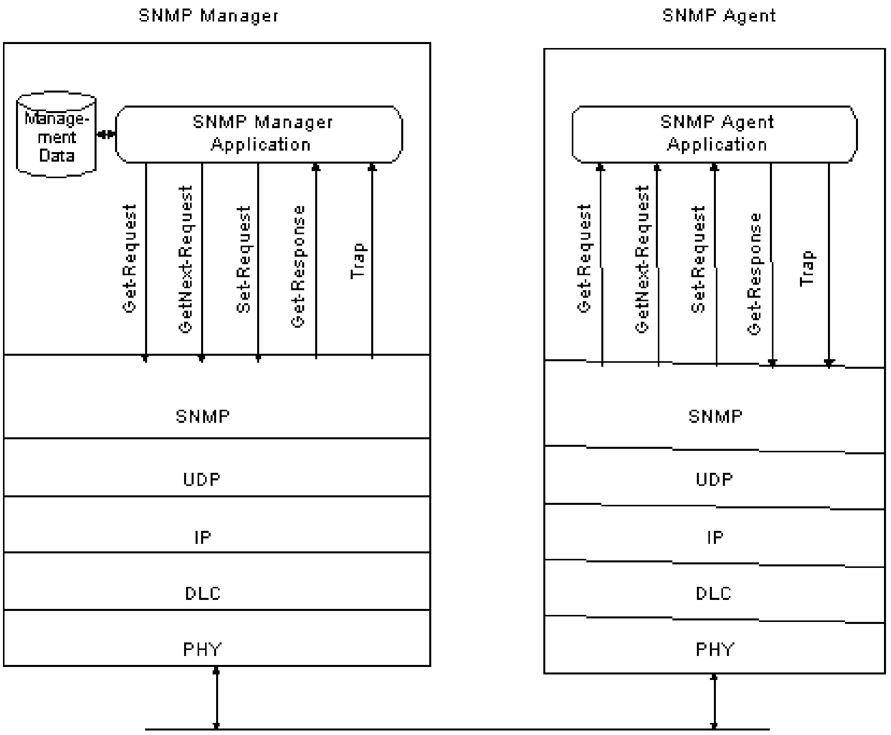


Figure 4.9 SNMP Network Management Architecture

- The communication of management information among management entities is realized through exchange of following 5 protocol messages:

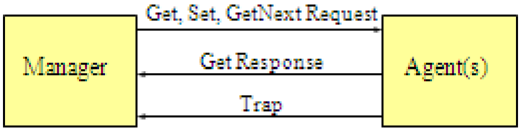
  1) The *get-request* message is generated by the management process requesting the value of an object.

  2) The *get-next-request* is similar to get-request. In many situations, an object may have multiple values because of multiple instances of the object.

  3) The *set-request* is generated by the management process to initialize or reset the value of an object variable.

  4) The *get-response* message is generated by an agent process. It is generated only on receipt of a get-request, get-next-request or set-request message from a management process.

  5) A *trap* is an unsolicited message generated by an agent process without a message or event arriving from the manager process.



- The SNMP manager has a database that polls the managed objects for management data. It contains 2 sets of data: one on the information about the objects, MIB and a second on the values of the objects, MDB

  1) A *MIB* is a virtual database and is static. In fact, a MIB needs to be there when an NMS discovers a new object in the network. It is compiled in the manager during the implementation.

  2) A *MDB* is dynamic and contains the measured values associated with the object. This is a true database. It is implemented using any database architecture chosen by the implementers.

UNIT 4: SNMPv1 NETWORK MANAGEMENT — ORGANIZATION & INFORMATION MODELS (CONT.)

SMI

- SMI stand for Structure of Management Information.
- A managed object can be considered to be composed of an object type and an object instance (fig:4.10).
- SMI is concerned only with the object type and not object instance. i.e. the object instance is not defined by SMI
- Object type, which is a data type, has following:

1) The name is represented uniquely by a descriptor and object identifier. For example,

```
internet OBJECT IDENTIFIER ::=
                    {iso org(3) dod(6) 1 }.
```

2) The syntax of an object type is defined using the ASN.1.

3) BER have been adopted as the encoding scheme for transfer of data types between agent and manager processes as well as between manager processes.

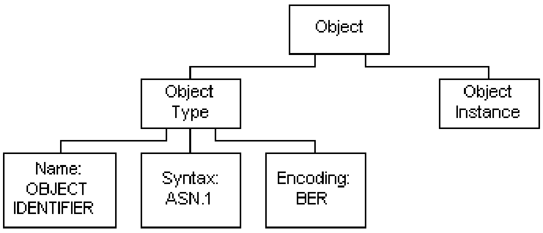


Figure 4.10 Managed Object : Type and Instance

- Every object type (i.e. every name) is uniquely identified by a DESCRIPTOR and an associated OBJECT IDENTIFIER. (fig:4.11).
- Internet MIB has its OBJECT IDENTIFIER 1.3.6.1, which can be defined as

internet OBJECT IDENTIFIER::{iso org(3) dod(6) 1}

- Any object in the Internet MIB will start with the prefix 1.3.6.1 or internet. For eg, there are 4 objects under the internet object (fig:4.13)

1) The directory(1) node i reserved for future use of OSI Directory in the Internet.

2) The mgmt(2) is used to identify all IETF-recommended and IAB-approved sub-nodes and objects.

3) The experimental(3) node was created to define objects under IETF experiments.

4) The private(4) node is a heavily used node. Commercial vendors can acquire a number under enterprises(1),which is under the private(4) node.

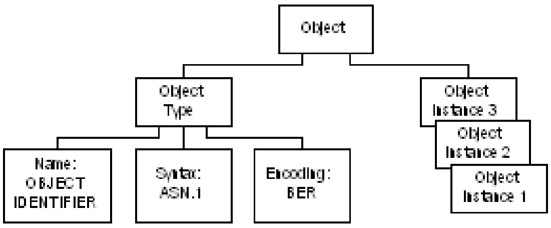


Figure 4.11 Managed Object : Type with Multiple Instances

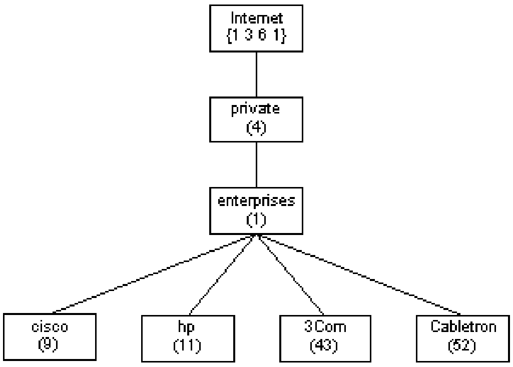


Figure 4.14 Private Subtree for Commercial Vendors

SNMP-BASED ASN.1 DATA TYPE STRUCTURE

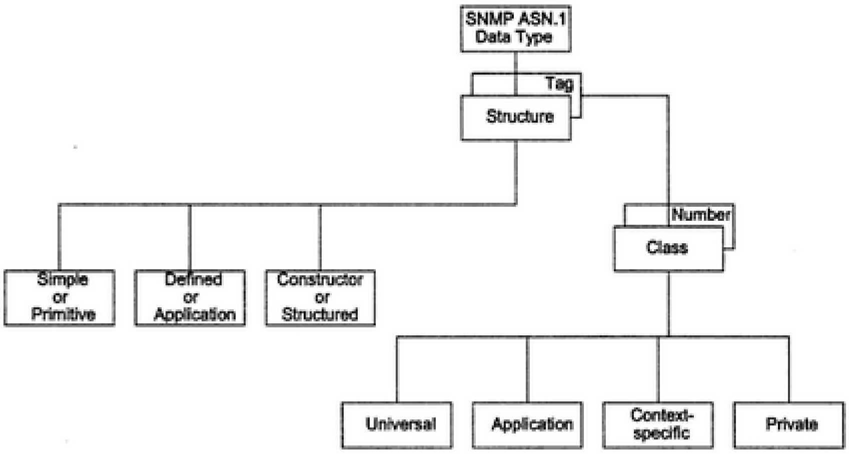
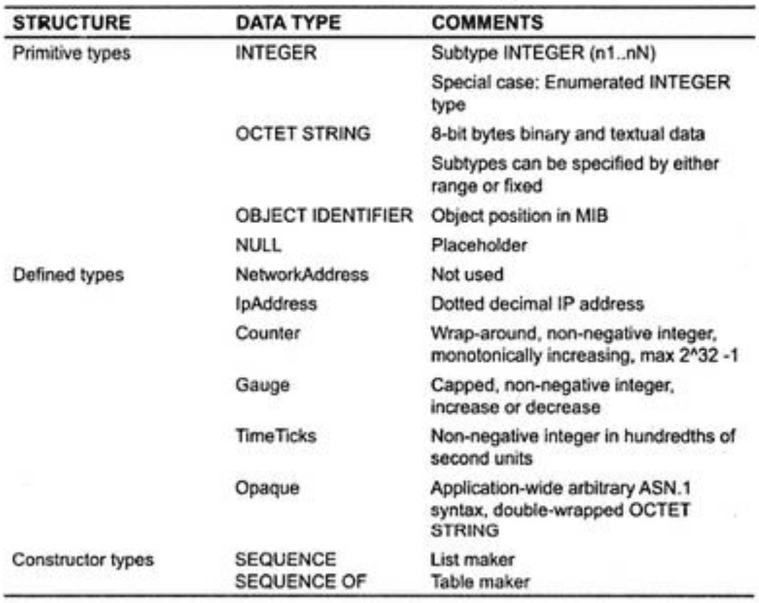


Figure 4.15 SNMP ASN.1 Data Type

Table 4.1 SNMP-based ASN.1 Data Type Structure



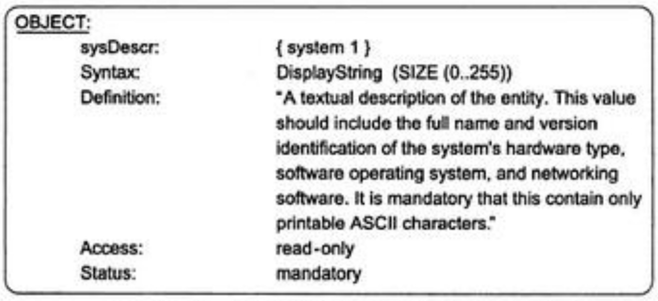
STRUCTURE	DATA TYPE	COMMENTS
Primitive types	INTEGER	Subtype INTEGER (n1..nN)
		Special case: Enumerated INTEGER type
	OCTET STRING	8-bit bytes binary and textual data
		Subtypes can be specified by either range or fixed
	OBJECT IDENTIFIER	Object position in MIB
	NULL	Placeholder
Defined types	NetworkAddress	Not used
	IpAddress	Dotted decimal IP address
	Counter	Wrap-around, non-negative integer, monotonically increasing, max 2^32 -1
	Gauge	Capped, non-negative integer, increase or decrease
	TimeTicks	Non-negative integer in hundredths of second units
	Opaque	Application-wide arbitrary ASN.1 syntax, double-wrapped OCTET STRING
Constructor types	SEQUENCE	List maker
	SEQUENCE OF	Table maker

- Defined are defined using primitive types. The primitive types used are NetworkAddress, IpAddress, Counter, Gauge and TimeTicks.
- NetworkAddress is a choice of the address of the protocol family. For eg, TCP/IP-based Internet family, which uses the base type IpAddress.
- IpAddress is the conventional four groups of dotted decimal notation of IPv4, for e.g. 190.146.252.255. The 32 bit string is designated as OCTET STRING of length 4,in network byte order.
- Counter is an application-wide data type and is a non-negative integer. It can only increase in value up to a maximum of 2^{32-1} and then wraps around starting from 0. Counter types is useful for defining values of data types that continually increase such as input packets received on an interface or output packet errors on an interface.
- Gauge is also a non-negative integer, but its value an move either up or down. It pegs at its maximum value of 2^{32-1}. Gauge is used for data types whose value increases or decreases such as the number of interfaces that are active in a router or hub.
- TimeTicks is a non-negative integer and measures time in units of hundredths of a second. Its value indicates in hundredths of a second the number of units of time between the current instant and the time it was initialized to 0. The maximum value is 2^{32-1}.
- Opaque is used to specify octets of binary information. It is an application-wide data type that supports the capability to pass arbitrary ASN.1 syntax. It is used to create data types based on previously defined data types. Its size is undefined in SNMPv1, which causes some problem in its implementation.

MANAGED OBJECTS

- A managed object has following 5 parameters:

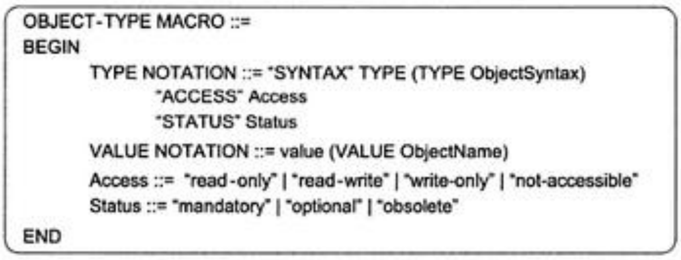
  1) The textual name for an object type is mnemonic and is defined as OBJECT DESCRIPTOR. OBJECT DESCRIPTOR defines only the object type and not the occurrence or instantiation of it. Associated with each OBJECT DESCRIPTOR is an OBJECT IDENTIFIER, which is the unique position it occupies in the MIB,
  2) Syntax is the ASN.1 definition of the object type,
  3) A definition is an accepted textual description of the object type. It is a basis for the common language, or semantics, to be used by all vendors. It is intended to avoid confusion in the exchange of information between the managed object and the management system as well as between the various network management systems,
  4) Access is the specification for the type of privilege associated with accessing the information: read-only, read-write or not-accessible. Its value is defined by the system vendor during the manufacturing process,
  5) Status specifies whether the managed object is current or obsolete. The 3 choices for status are mandatory, optional and obsolete. A managed object, once defined, can only be made obsolete and not removed or deleted. If it is current, the implementation of it is specified as either mandatory or optional,



```
OBJECT:
        sysDescr:        { system 1 }
        Syntax:          DisplayString  (SIZE (0..255))
        Definition:      "A textual description of the entity. This value
                         should include the full name and version
                         identification of the system's hardware type,
                         software operating system, and networking
                         software. It is mandatory that this contain only
                         printable ASCII characters."
        Access:          read-only
        Status:          mandatory
```

Figure 4.17 Specifications for System Description

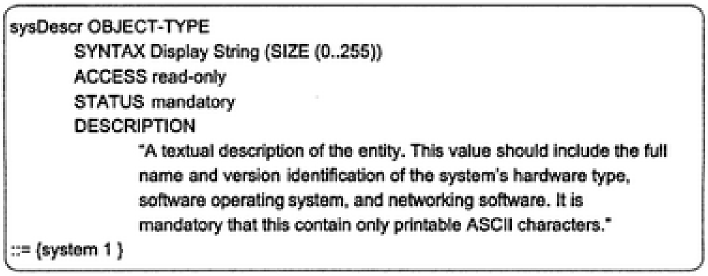
MACROS FOR MANAGED OBJECTS



```
OBJECT-TYPE MACRO ::=
BEGIN
        TYPE NOTATION ::= "SYNTAX" TYPE (TYPE ObjectSyntax)
                "ACCESS" Access
                "STATUS" Status
        VALUE NOTATION ::= value (VALUE ObjectName)
        Access ::=  "read-only" | "read-write" | "write-only" | "not-accessible"
        Status ::= "mandatory" | "optional" | "obsolete"
END
```

(a) An OBJECT-TYPE Macro [RFC 1155]

Figure 4.18 Scalar OBJECT-TYPE Macro and Example



```
sysDescr OBJECT-TYPE
        SYNTAX Display String (SIZE (0..255))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
                "A textual description of the entity. This value should include the full
                name and version identification of the system's hardware type,
                software operating system, and networking software. It is
                mandatory that this contain only printable ASCII characters."
::= {system 1 }
```

(b) A Scalar or Single Instance Macro: sysDescr [RFC 1213]

Figure 4.18 (continued)

- The body of the macro module consists of 3 parts: type notation, value notation and supporting productions.

  1) *TYPE NOTATION* defines the object types in the module and *VALUE NOTATION* defines the name of the object,
  2) Access can be only one of 4 options: read-only, read-write, write-only or not-accessible,
  3) Allowed values for Status are mandatory, optional or obsolete,

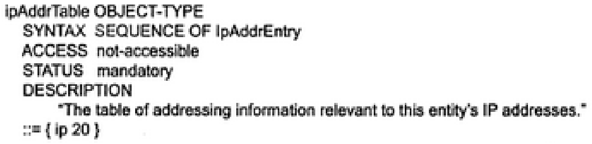
AGGREGATE OBJECT

- A group of objects,
- Also called tabular objects,
- Can be represented by a table with
  - → Columns of objects
  - → Rows of instances
- Example: IP address table
- Consists of objects:
  - → IP address
  - → Interface
  - → Subnet mask (which subnet this address belongs to)
  - → Broadcast address (value of l.s.b. in IP broadcast address)
  - → Largest IP datagram that can be assembled
- Multiple instances of these objects associated with the node.

AGGREGATE MANAGED OBJECT MACRO

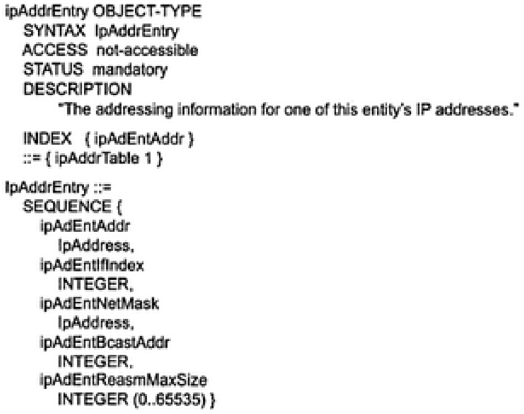
- Index ipAdEntAddr uniquely identifies an instance.
- May require more than one object in the instance to uniquely identify it.

Table Object



```
ipAddrTable OBJECT-TYPE
   SYNTAX  SEQUENCE OF IpAddrEntry
   ACCESS  not-accessible
   STATUS   mandatory
   DESCRIPTION
        "The table of addressing information relevant to this entity's IP addresses."
   ::= { ip 20 }
```

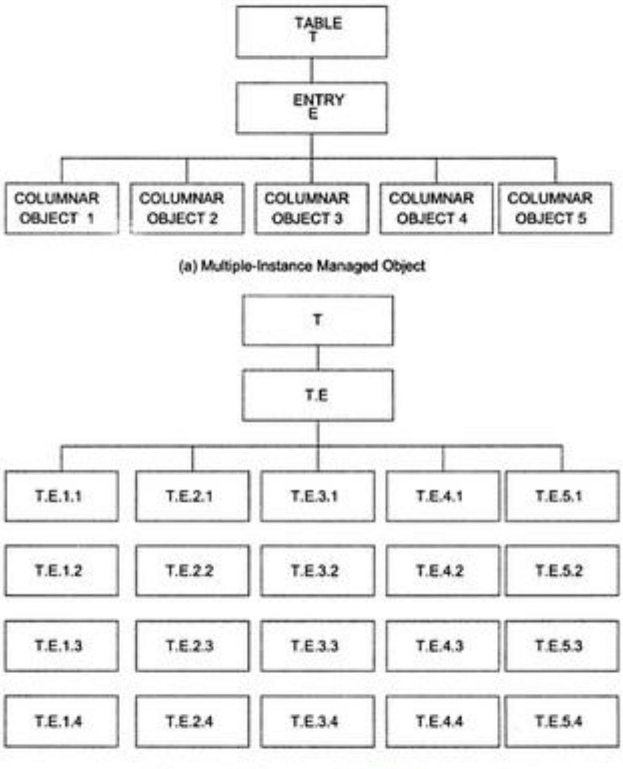
Entry Object



```
ipAddrEntry OBJECT-TYPE
   SYNTAX  IpAddrEntry
   ACCESS  not-accessible
   STATUS  mandatory
   DESCRIPTION
        "The addressing information for one of this entity's IP addresses."

   INDEX   { ipAdEntAddr }
   ::= { ipAddrTable 1 }

IpAddrEntry ::=
   SEQUENCE {
      ipAdEntAddr
         IpAddress,
      ipAdEntIfIndex
         INTEGER,
      ipAdEntNetMask
         IpAddress,
      ipAdEntBcastAddr
         INTEGER,
      ipAdEntReasmMaxSize
         INTEGER (0..65535) }
```

TABULAR REPRESENTATION OF AGGREGATE OBJECT

- The objects TABLE T and ENTRY E are objects that 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 objects1 through 5.
- Multiple instances of E are represented by multiple rows.
- Notice that the column-row numeric designation is reverse of what we are used to as row-column.



(b) Example of a 5-Columnar Object with 4 Instances (Rows)

Figure 4.22 Numbering Convention of a Managed Object Table

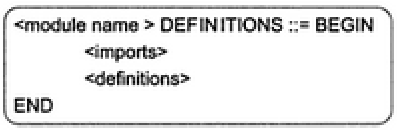
MIB

- MIB stand for Management Information Base.
- This is a virtual information base. Managed objects are accessed via this virtual information base.
- Objects in the MIB are defined using ASN.1. The objects defined in MIB-2 have the OBJECT IDENTIFIER prefix:

mib-2 OBJECT IDENTIFIER ::= {mgmt 1}

OBJECT GROUPS

- Objects that are related are grouped into object groups.
- Object groups facilitate logical assignment of object identifiers.
- One of the criteria for choosing objects to be included in standards is that the object is essential for either fault or configuration management.

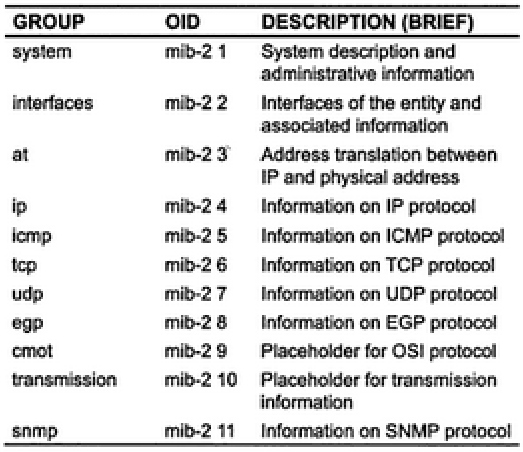


```
<module name > DEFINITIONS ::= BEGIN
          <imports>
          <definitions>
END
```

Figure 4.25 MIB Module Structure

- 11 groups are defined in MIB2. (Fig:4.26).

Table 4.4 MIB-II Groups



GROUP	OID	DESCRIPTION (BRIEF)
system	mib-2 1	System description and administrative information
interfaces	mib-2 2	Interfaces of the entity and associated information
at	mib-2 3	Address translation between IP and physical address
ip	mib-2 4	Information on IP protocol
icmp	mib-2 5	Information on ICMP protocol
tcp	mib-2 6	Information on TCP protocol
udp	mib-2 7	Information on UDP protocol
egp	mib-2 8	Information on EGP protocol
cmot	mib-2 9	Placeholder for OSI protocol
transmission	mib-2 10	Placeholder for transmission information
snmp	mib-2 11	Information on SNMP protocol

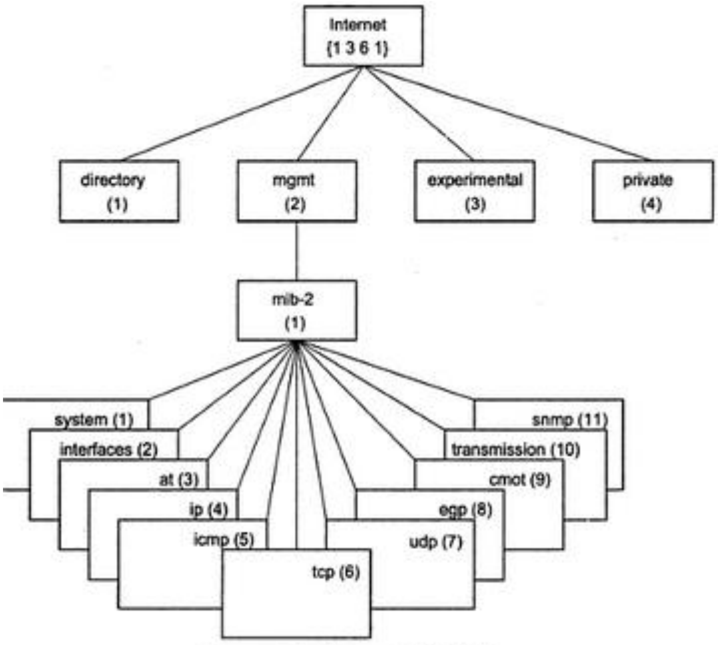
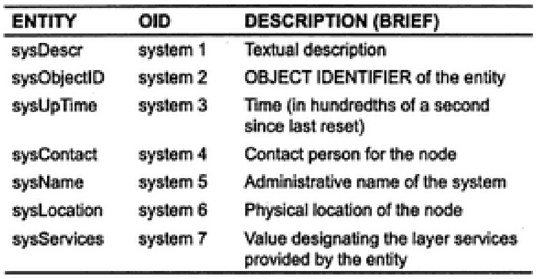


Figure 4.26 Internet MIB-II Group

SYSTEM GROUP

- The System group is the basic group in the Internet standard MIB (fig:4.27).
- Its elements are probably the most accessed managed objects.
- After an NMS discovers all the components in a network or the new components in the network,it has to obtain information on the system it discovered such as system name,object ID and so on.
- The NMS will initiate the get-request message on the objects in this group for this purpose.
- The group also has administrative information such as contact peron and physical location, that helps a network manager.
- Implementation of the System group is mandatory for all systems in both agent & manager.

Table 4.5 System Group



ENTITY	OID	DESCRIPTION (BRIEF)
sysDescr	system 1	Textual description
sysObjectID	system 2	OBJECT IDENTIFIER of the entity
sysUpTime	system 3	Time (in hundredths of a second since last reset)
sysContact	system 4	Contact person for the node
sysName	system 5	Administrative name of the system
sysLocation	system 6	Physical location of the node
sysServices	system 7	Value designating the layer services provided by the entity

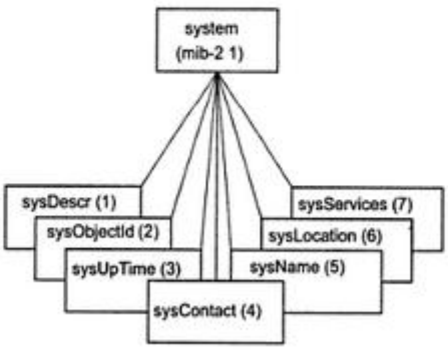
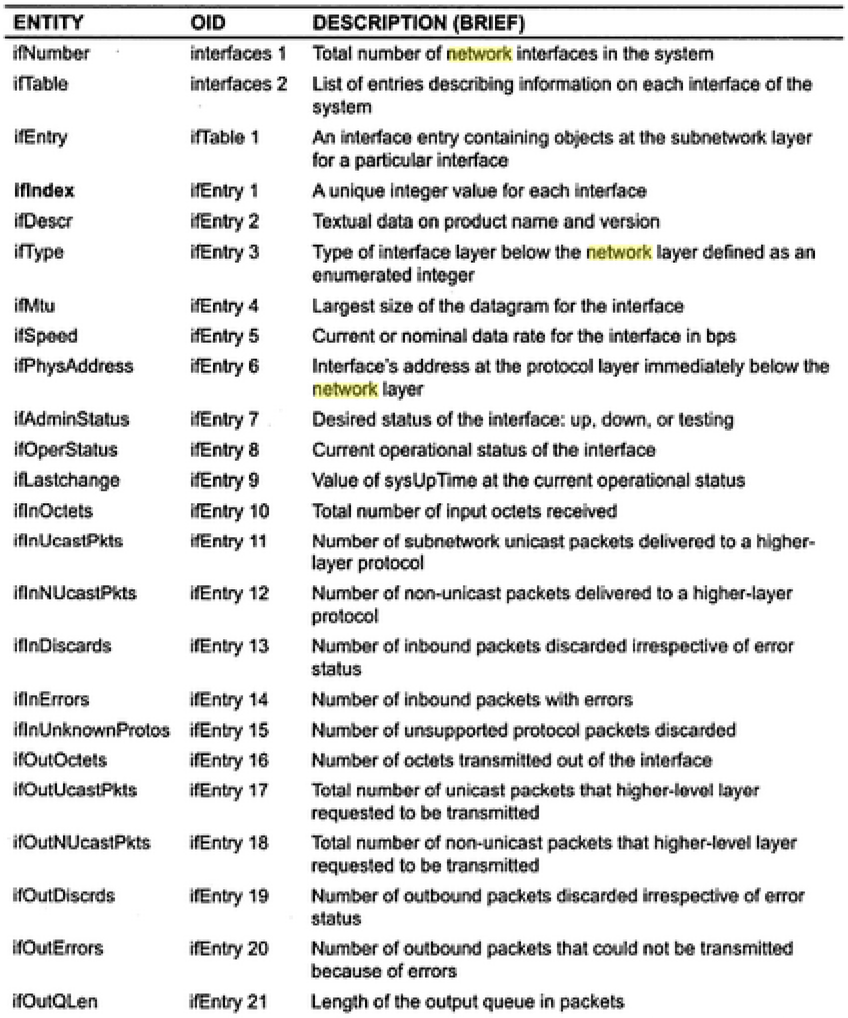


Figure 4.27 System Group

INTERFACES GROUP

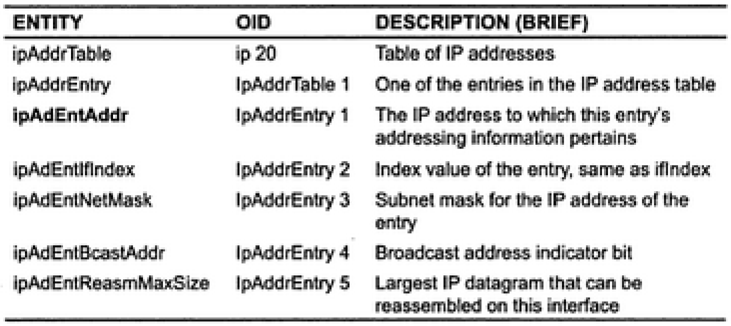
- The Interface group contains managed objects associated with the interfaces of a system.
- If there is more than one interface in the system, the group describes the parameters associated with each interface
- This specifies the number of interfaces in a network component and the managed objects associated with each interface.
- Implementation of the Interfaces group is mandatory for all system.

Table 4.6 Interfaces Group



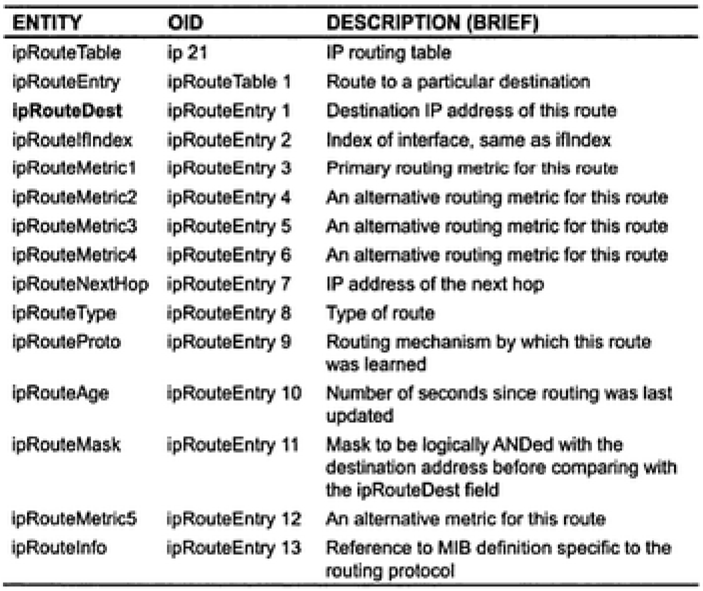
ENTITY	OID	DESCRIPTION (BRIEF)
ifNumber	interfaces 1	Total number of network interfaces in the system
ifTable	interfaces 2	List of entries describing information on each interface of the system
ifEntry	ifTable 1	An interface entry containing objects at the subnetwork layer for a particular interface
IfIndex	ifEntry 1	A unique integer value for each interface
ifDescr	ifEntry 2	Textual data on product name and version
ifType	ifEntry 3	Type of interface layer below the network layer defined as an enumerated integer
ifMtu	ifEntry 4	Largest size of the datagram for the interface
ifSpeed	ifEntry 5	Current or nominal data rate for the interface in bps
ifPhysAddress	ifEntry 6	Interface's address at the protocol layer immediately below the network layer
ifAdminStatus	ifEntry 7	Desired status of the interface: up, down, or testing
ifOperStatus	ifEntry 8	Current operational status of the interface
ifLastchange	ifEntry 9	Value of sysUpTime at the current operational status
ifInOctets	ifEntry 10	Total number of input octets received
ifInUcastPkts	ifEntry 11	Number of subnetwork unicast packets delivered to a higher-layer protocol
ifInNUcastPkts	ifEntry 12	Number of non-unicast packets delivered to a higher-layer protocol
ifInDiscards	ifEntry 13	Number of inbound packets discarded irrespective of error status
ifInErrors	ifEntry 14	Number of inbound packets with errors
ifInUnknownProtos	ifEntry 15	Number of unsupported protocol packets discarded
ifOutOctets	ifEntry 16	Number of octets transmitted out of the interface
ifOutUcastPkts	ifEntry 17	Total number of unicast packets that higher-level layer requested to be transmitted
ifOutNUcastPkts	ifEntry 18	Total number of non-unicast packets that higher-level layer requested to be transmitted
ifOutDiscrds	ifEntry 19	Number of outbound packets discarded irrespective of error status
ifOutErrors	ifEntry 20	Number of outbound packets that could not be transmitted because of errors
ifOutQLen	ifEntry 21	Length of the output queue in packets

Table 4.8 IP Address Table



ENTITY	OID	DESCRIPTION (BRIEF)
ipAddrTable	ip 20	Table of IP addresses
ipAddrEntry	IpAddrTable 1	One of the entries in the IP address table
ipAdEntAddr	IpAddrEntry 1	The IP address to which this entry's addressing information pertains
ipAdEntIfIndex	IpAddrEntry 2	Index value of the entry, same as ifIndex
ipAdEntNetMask	IpAddrEntry 3	Subnet mask for the IP address of the entry
ipAdEntBcastAddr	IpAddrEntry 4	Broadcast address indicator bit
ipAdEntReasmMaxSize	IpAddrEntry 5	Largest IP datagram that can be reassembled on this interface

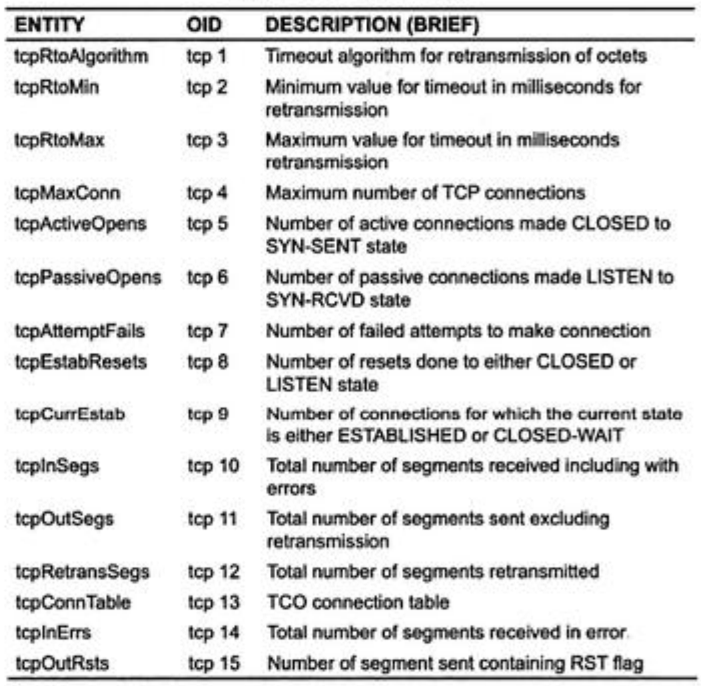
Table 4.9 IP Routing Table



ENTITY	OID	DESCRIPTION (BRIEF)
ipRouteTable	ip 21	IP routing table
ipRouteEntry	ipRouteTable 1	Route to a particular destination
ipRouteDest	ipRouteEntry 1	Destination IP address of this route
ipRouteIfIndex	ipRouteEntry 2	Index of interface, same as ifIndex
ipRouteMetric1	ipRouteEntry 3	Primary routing metric for this route
ipRouteMetric2	ipRouteEntry 4	An alternative routing metric for this route
ipRouteMetric3	ipRouteEntry 5	An alternative routing metric for this route
ipRouteMetric4	ipRouteEntry 6	An alternative routing metric for this route
ipRouteNextHop	ipRouteEntry 7	IP address of the next hop
ipRouteType	ipRouteEntry 8	Type of route
ipRouteProto	ipRouteEntry 9	Routing mechanism by which this route was learned
ipRouteAge	ipRouteEntry 10	Number of seconds since routing was last updated
ipRouteMask	ipRouteEntry 11	Mask to be logically ANDed with the destination address before comparing with the ipRouteDest field
ipRouteMetric5	ipRouteEntry 12	An alternative metric for this route
ipRouteInfo	ipRouteEntry 13	Reference to MIB definition specific to the routing protocol

TCP GROUP

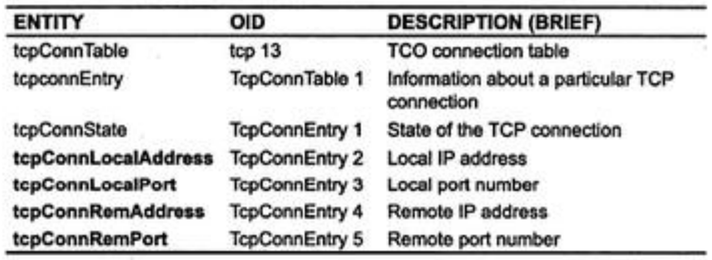
Table 4.13 TCP Group



ENTITY	OID	DESCRIPTION (BRIEF)
tcpRtoAlgorithm	tcp 1	Timeout algorithm for retransmission of octets
tcpRtoMin	tcp 2	Minimum value for timeout in milliseconds for retransmission
tcpRtoMax	tcp 3	Maximum value for timeout in milliseconds retransmission
tcpMaxConn	tcp 4	Maximum number of TCP connections
tcpActiveOpens	tcp 5	Number of active connections made CLOSED to SYN-SENT state
tcpPassiveOpens	tcp 6	Number of passive connections made LISTEN to SYN-RCVD state
tcpAttemptFails	tcp 7	Number of failed attempts to make connection
tcpEstabResets	tcp 8	Number of resets done to either CLOSED or LISTEN state
tcpCurrEstab	tcp 9	Number of connections for which the current state is either ESTABLISHED or CLOSED-WAIT
tcpInSegs	tcp 10	Total number of segments received including with errors
tcpOutSegs	tcp 11	Total number of segments sent excluding retransmission
tcpRetransSegs	tcp 12	Total number of segments retransmitted
tcpConnTable	tcp 13	TCO connection table
tcpInErrs	tcp 14	Total number of segments received in error
tcpOutRsts	tcp 15	Number of segment sent containing RST flag

TCP CONNECTION TABLE

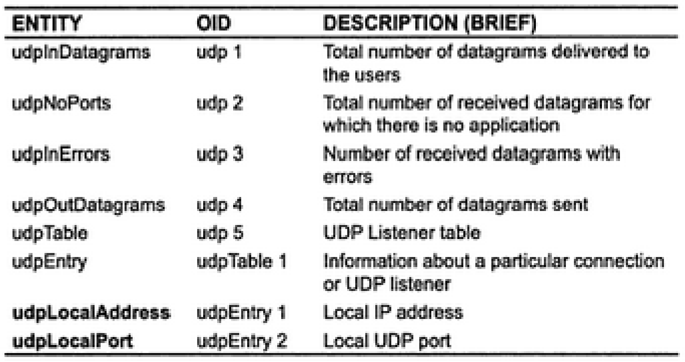
Table 4.14 TCP Connection Table



ENTITY	OID	DESCRIPTION (BRIEF)
tcpConnTable	tcp 13	TCO connection table
tcpconnEntry	TcpConnTable 1	Information about a particular TCP connection
tcpConnState	TcpConnEntry 1	State of the TCP connection
tcpConnLocalAddress	TcpConnEntry 2	Local IP address
tcpConnLocalPort	TcpConnEntry 3	Local port number
tcpConnRemAddress	TcpConnEntry 4	Remote IP address
tcpConnRemPort	TcpConnEntry 5	Remote port number

UDP GROUP

Table 4.15 UDP Group



ENTITY	OID	DESCRIPTION (BRIEF)
udpInDatagrams	udp 1	Total number of datagrams delivered to the users
udpNoPorts	udp 2	Total number of received datagrams for which there is no application
udpInErrors	udp 3	Number of received datagrams with errors
udpOutDatagrams	udp 4	Total number of datagrams sent
udpTable	udp 5	UDP Listener table
udpEntry	udpTable 1	Information about a particular connection or UDP listener
udpLocalAddress	udpEntry 1	Local IP address
udpLocalPort	udpEntry 2	Local UDP port

UNIT 4(CONT.): SNMPv1 NETWORK MANAGEMENT — COMMUNICATION & FUNCTIONAL MODELS

SNMP COMMUNICATION ARCHITECTURE

- The SNMP architecture consists of communication between network management stations and managed network elements. (Fig:4.9).
- Network elements have built-in management agents if they are managed elements.
- The SNMP communication protocol is used to communicate information between the network management stations and the management agents in the elements.
- Only non-aggregate objects are communicated using SNMP. The aggregate objects are communicated as instances of the object.
- ASN.1 and BER are used for data transfer in SNMP.
- The information about the network is obtained primarily by the management stations polling the agents.
- The SNMP manages the network with the following 5 messages:

  1) The *get-request* message is generated by the management process requesting the value of an object.

  2) The *get-next-request* is similar to get-request. In many situations, an object may have multiple values because of multiple instances of the object.

  3) The *set-request* is generated by the management process to initialize or reset the value of an object variable.

  4) The *get-response* message is generated by an agent process. It is generated only on receipt of a get-request, get-next-request or set-request message from a management process.

  5) A *trap* is an unsolicited message generated by an agent process without a message or event arriving from the manager process.
- Following are 3 types of traps:

  1) The *generic trap* type consists of coldStart, warmStart, linkDown, linkUp, authenticationFailure, egpNeighbourLoss and enterpriseSpecific.

  2) The *specific-trap* is a specific code and is generated even when an enterpriseSpecific trap is not present.

  3) The *time-stamp* is the time elapsed between the last initialization or re-initialization of the element and the generation of the trap.

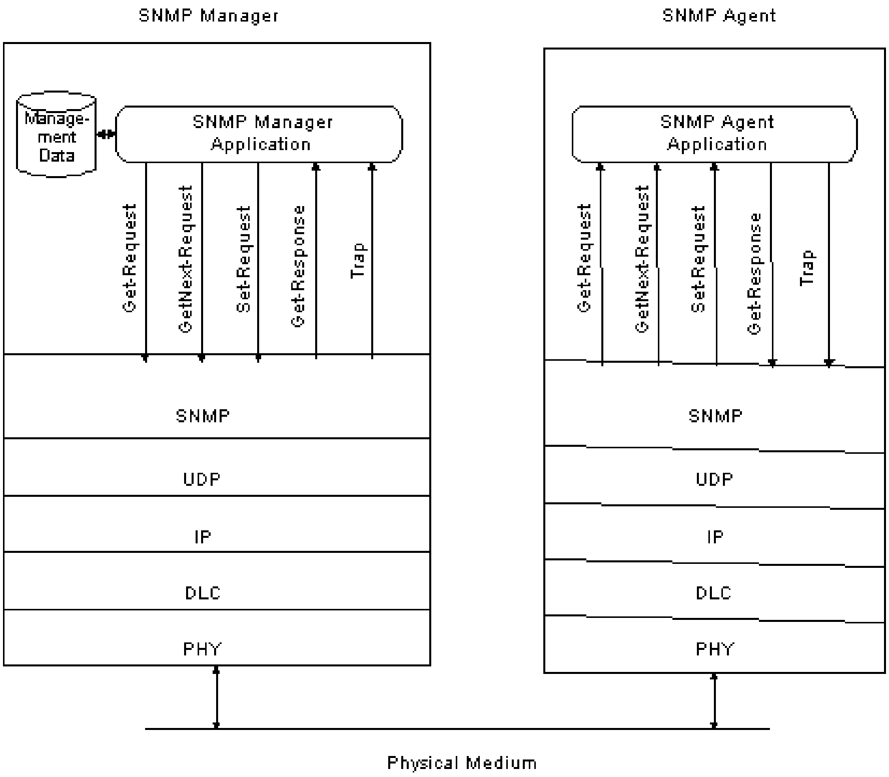


Figure 4.9 SNMP Network Management Architecture

ADMINISTRATIVE MODEL

- The application entity that reside in the management station is called SNMP manager, and the application entity that reside in the network element is called SNMP agent.. The pairing of these two entities is called an SNMP community.
- In (fig:5.1),while an SNMP manager is monitoring traffic on an element, another manager can be configuring some administrative information on it. A third manager can be monitoring it to perform some statistical study.
- In (fig:5.1),the authentication scheme is filter module in the manager and in the agent. The simplest form of authentication is the common community name between the two application entities.
- A network element comprises many managed objects, both standard & private. However,a management agent may be permitted to view only a subset of the network element's managed objects. This is called the community MIB view (fig:5.2).
- In addition to the MIB view, each community name is also assigned an SNMP access mode either READ-ONLY or READ-WRITE.
- The pairing of the SNMP MIB view with the SNMP access mode is called the community profile.
- A community profile in combination with the access mode of a managed object determines the operation that can be performed on the object by an agent.

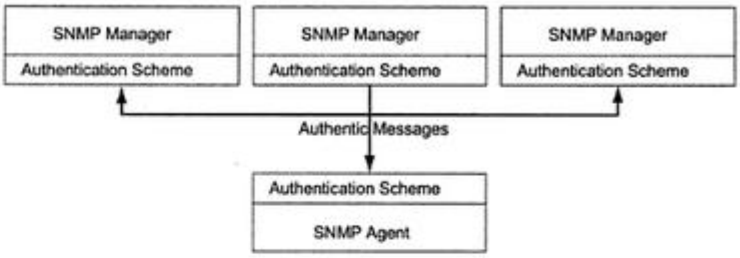


Figure 5.1 SNMP Community

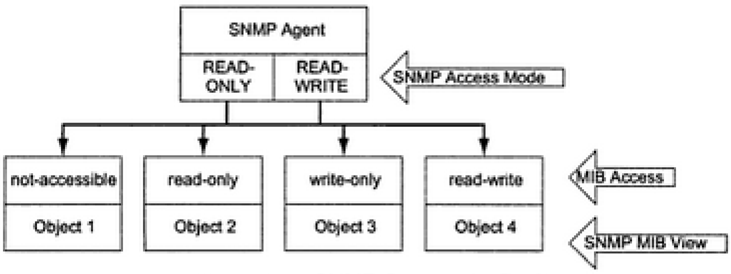


Figure 5.2 SNMP Community Profile

SNMP ACCESS POLICY

- A pairing of an SNMP community with an SNMP community profile is defined as SNMP access policy. This defines the administrative model of SNMP management.
- In fig:5.3, agent 1 and 2 belong to Community 1. However, they have different community profiles, community profiles 1 and 2.
- Manager 1, which is part of Community 1, can communicate with both Agents 1 and 2. However, it cannot communicate with Agents 3 and 4, which belong to Community 2. Manager 2 has access to them because it also belongs to Community 2.
- The SNMP access policy can be extended to managing a non-SNMP community that uses the SNMP proxy access policy (fig:5.4).
- The SNMP agent associated with the proxy policy is called a proxy agent or commercially a proxy server.
- The proxy agent monitors a non-SNMP community with non-SNMP agent and then converts the objects and data to SNMP-compatible objects and data to feed to an SNMP manager.

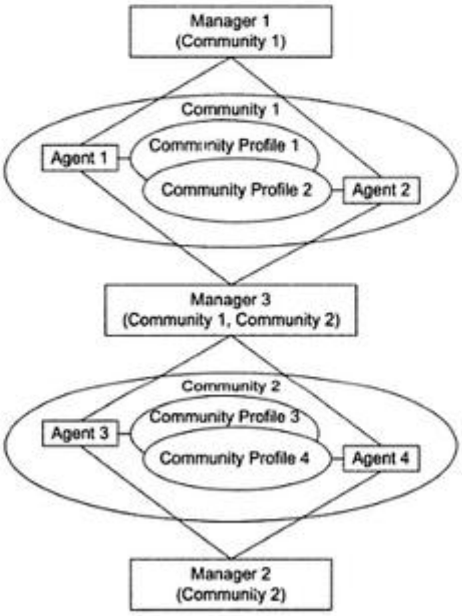


Figure 5.3 SNMP Access Policy

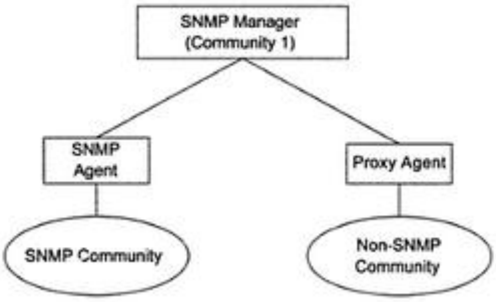


Figure 5.4 SNMP Proxy Access Policy

SNMP PROTOCOL SPECIFICATIONS

- The peer processes, which implement the SNMP, and thus support the SNMP application entities, are called protocol entities.
- Communication among protocol entities is accomplished using messages encapsulated in UDP datagrams.
- An SNMP message consists of a version identifier, an SNMP community name and a PDU (fig:5.5).
- An SNMP protocol entity is received on port 161 on the host except for trap, which is received on port 16a
- The maximum length of the protocol in SNMPv1 is 484 bytes.
- This is mandatory that all five PDUs be supported in all implementations:GetRequest-PDU,GetNextRequest-PDU,GetResponse-PDU,SetRequest-PDU and Trap-PDU.
- Basic operations of the protocol entity involve the following steps as a guide to implementation:

1) The protocol entity that generates the message constructs the appropriate data PDU as an ASN.1 object.
2) It then passes the ASN.1 object, along with a community name and the transport addresses of itself and the destination ,to the authentication scheme.
3) The authentication scheme returns another ASN.1 object.
4) The protocol entity now constructs the message to be transmitted with the version number, community name and the new ASN.1 object, then serializes it using the BER, and transmits it.
5) The reverse process goes on at the receiver.
6) The message is discarded if error is encountered in any of the steps.
7) A trap may be generated in case of authentication failure.
8) On successful receipt of the message, a return message is generated, if the original message is a get or set message.

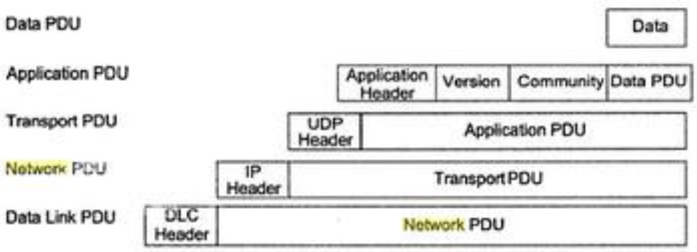
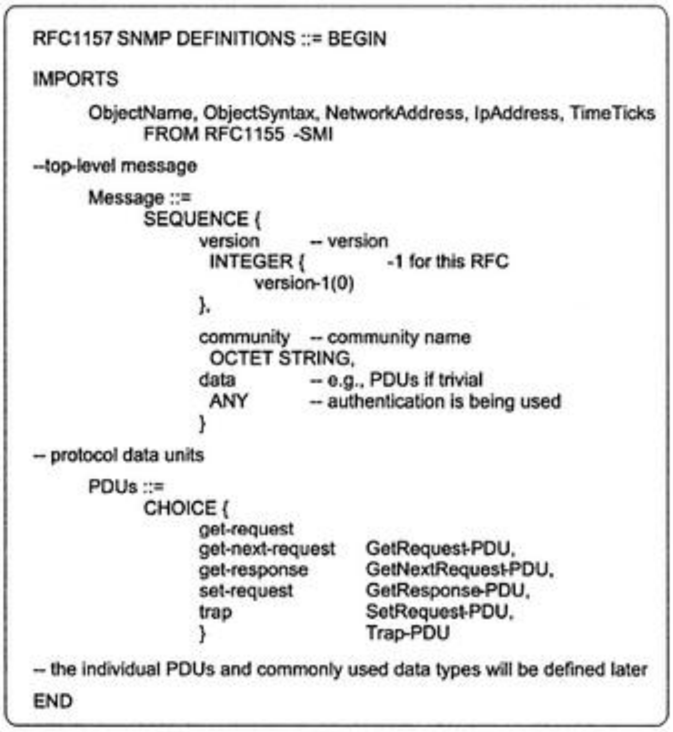


Figure 5.5 Encapsulated SNMP Message

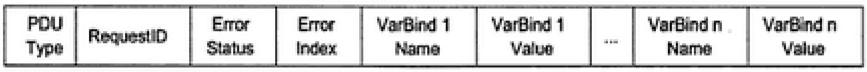


```
RFC1157 SNMP DEFINITIONS ::= BEGIN

IMPORTS
        ObjectName, ObjectSyntax, NetworkAddress, IpAddress, TimeTicks
                FROM RFC1155 -SMI

--top-level message
        Message ::=
                SEQUENCE {
                        version         -- version
                          INTEGER {             -1 for this RFC
                                version-1(0)
                        },

                        community       -- community name
                          OCTET STRING,
                        data            -- e.g., PDUs if trivial
                          ANY           -- authentication is being used
                        }

-- protocol data units
        PDUs ::=
                CHOICE {
                        get-request
                        get-next-request        GetRequest-PDU,
                        get-response            GetNextRequest-PDU,
                        set-request             GetResponse-PDU,
                        trap                    SetRequest-PDU,
                        }                       Trap-PDU

-- the individual PDUs and commonly used data types will be defined later
END
```

Figure 5.6 RFC 1157-SNMP Macro

GET AND SET TYPE PDUS

- In fig:5.8, RequestID is used to track a message with the expected response or indicate loss of the message. Loss-of-message detection is implementation specific, such as time out if no response is received for a request within a given time.
- ErrorStatus is used to indicate that an error occurred.
- ErrorIndex is used to provide additional information on the error status. The value is filled with NULL in cases where it is not applicable. Otherwise, it is filled with the varBind number where the error occurred.

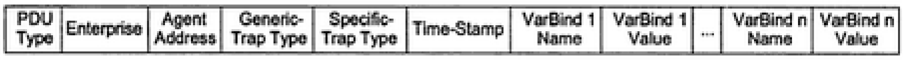


PDU Type	RequestID	Error Status	Error Index	VarBind 1 Name	VarBind 1 Value	...	VarBind n Name	VarBind n Value

Figure 5.8 Get and Set Type PDUs

TRAP PDU

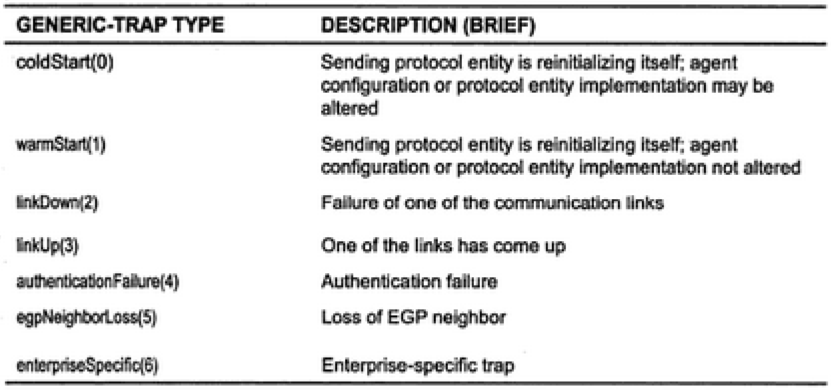
- In fig:5.9, the enterprise and agent-address pertain to the system generating the trap.
- The generic-trap consists of following 7 types:
  - → coldStart(0):sending protocol entity is reinitializing itself, agent's configuration or protocol entity implementation may be altered.
  - → warmStart(1):sending protocol entity is reinitializing itself, agent's configuration or protocol entity implementation not altered.
  - → linkDown(2):failure of one of the communication links.
  - → linkup(3):one of the links has come up.
  - → authenticationFailure(4):authentication failure.
  - → egpNeighborLoss(5):loss of EGP neighbor.
  - → enterpriseSpecific(6):Enterprise-specific trap.
- The integer in parenthesis associated with each name indicates the enumerated INTEGER.
- The specific-trap is a trap that is not covered by the enterpriseSpecific trap.
- Time-stamp indicates the elapsed time since last re-initialization.



PDU Type	Enterprise	Agent Address	Generic-Trap Type	Specific-Trap Type	Time-Stamp	VarBind 1 Name	VarBind 1 Value	...	VarBind n Name	VarBind n Value

Figure 5.9 Trap PDU

Table 5.1 Generic Traps



GENERIC-TRAP TYPE	DESCRIPTION (BRIEF)
coldStart(0)	Sending protocol entity is reinitializing itself; agent configuration or protocol entity implementation may be altered
warmStart(1)	Sending protocol entity is reinitializing itself; agent configuration or protocol entity implementation not altered
linkDown(2)	Failure of one of the communication links
linkUp(3)	One of the links has come up
authenticationFailure(4)	Authentication failure
egpNeighborLoss(5)	Loss of EGP neighbor
enterpriseSpecific(6)	Enterprise-specific trap

GETREQUEST-PDU OPERATION

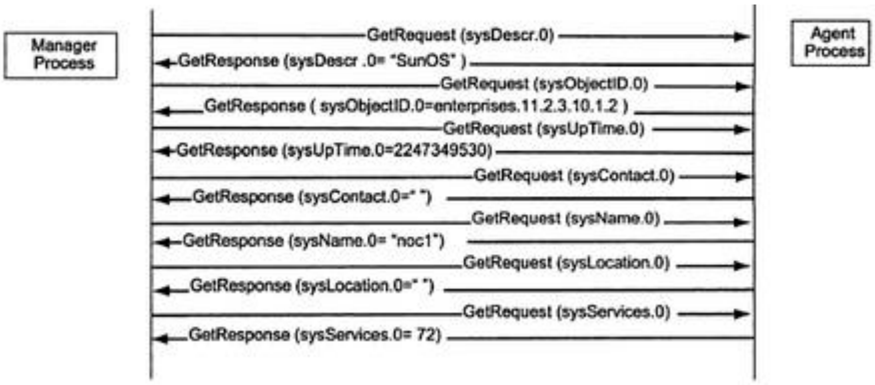

Manager Process
Agent Process
GetRequest (sysDescr.0)
GetResponse (sysDescr .0= "SunOS")
GetRequest (sysObjectID.0)
GetResponse (sysObjectID.0=enterprises.11.2.3.10.1.2)
GetRequest (sysUpTime.0)
GetResponse (sysUpTime.0=2247349530)
GetRequest (sysContact.0)
GetResponse (sysContact.0=" ")
GetRequest (sysName.0)
GetResponse (sysName.0= "noc1")
GetRequest (sysLocation.0)
GetResponse (sysLocation.0=" ")
GetRequest (sysServices.0)
GetResponse (sysServices.0= 72)

Figure 5.10 Get-Request Operation for System Group

GETNEXTREQUEST-PDU OPERATION

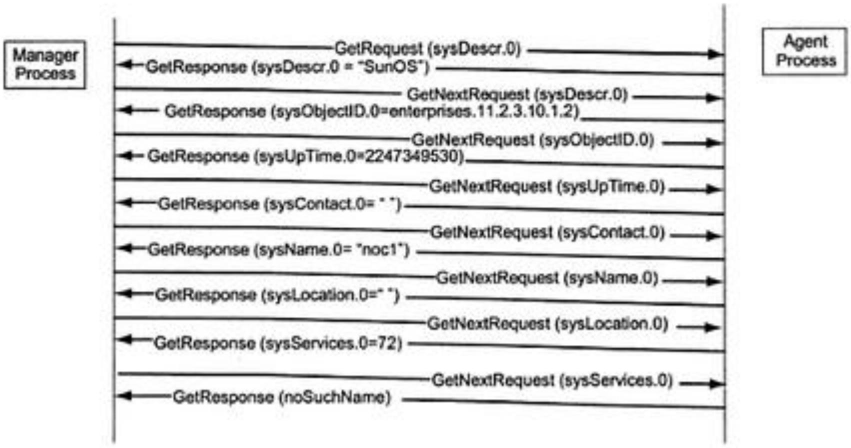

Manager Process
Agent Process
GetRequest (sysDescr.0)
GetResponse (sysDescr.0 = "SunOS")
GetNextRequest (sysDescr.0)
GetResponse (sysObjectID.0=enterprises.11.2.3.10.1.2)
GetNextRequest (sysObjectID.0)
GetResponse (sysUpTime.0=2247349530)
GetNextRequest (sysUpTime.0)
GetResponse (sysContact.0= " ")
GetNextRequest (sysContact.0)
GetResponse (sysName.0= "noc1")
GetNextRequest (sysName.0)
GetResponse (sysLocation.0=" ")
GetNextRequest (sysLocation.0)
GetResponse (sysServices.0=72)
GetNextRequest (sysServices.0)
GetResponse (noSuchName)

Figure 5.11 Get-Next-Request Operation for a System Group

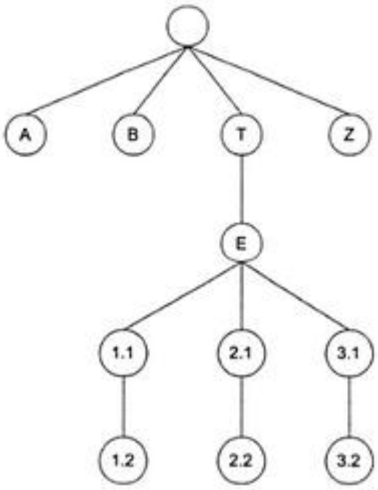
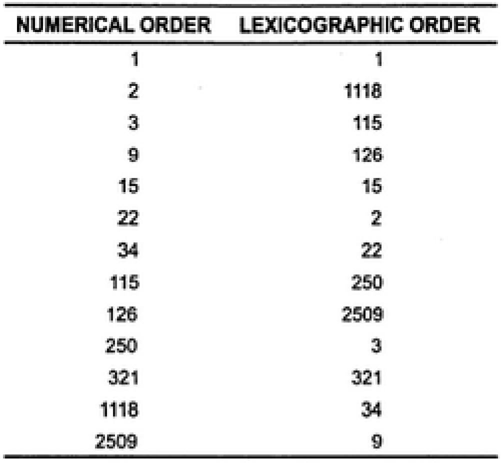


Figure 5.12 MIB for Operation Examples in Figures 5.13 and 5.15

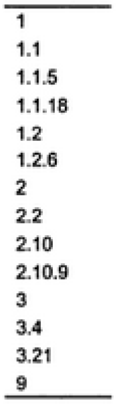
LEXICOGRAPHIC ORDER

Table 5.2 Lexicographic-Order Number Example



NUMERICAL ORDER	LEXICOGRAPHIC ORDER
1	1
2	1118
3	115
9	126
15	15
22	2
34	22
115	250
126	2509
250	3
321	321
1118	34
2509	9

Table 5.3 MIB Example for Lexicographic Ordering



1
1.1
1.1.5
1.1.18
1.2
1.2.6
2
2.2
2.10
2.10.9
3
3.4
3.21
9

- Procedure for ordering:
  1) Start with leftmost digit as first position.
  2) Before increasing the order in the first position, select the lowest digit in the second position.
  3) Continue the process till the lowest digit in the last position is captured.
  4) Increase the order in the last position until all the digits in the last position are captured.
  5) Move back to the last but one position and repeat the process.
  6) Continue advancing to the first position until all the numbers are ordered.
- Tree structure for the above process is shown below:

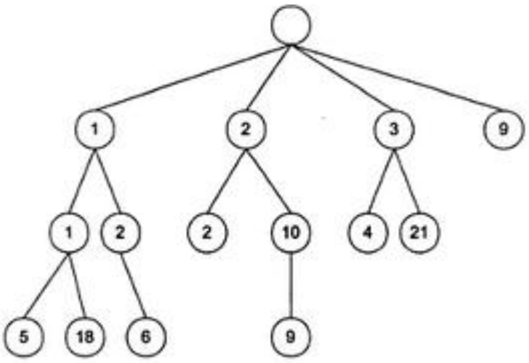


Figure 5.14 MIB Example for Lexicographic Ordering

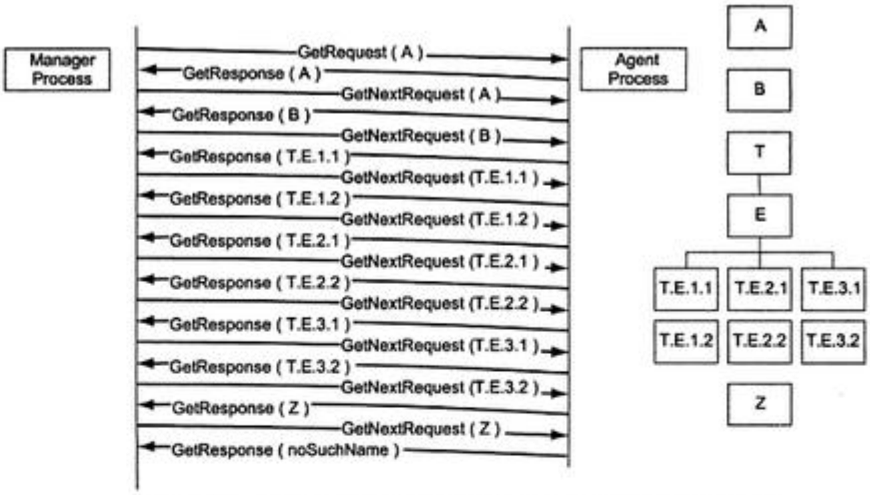
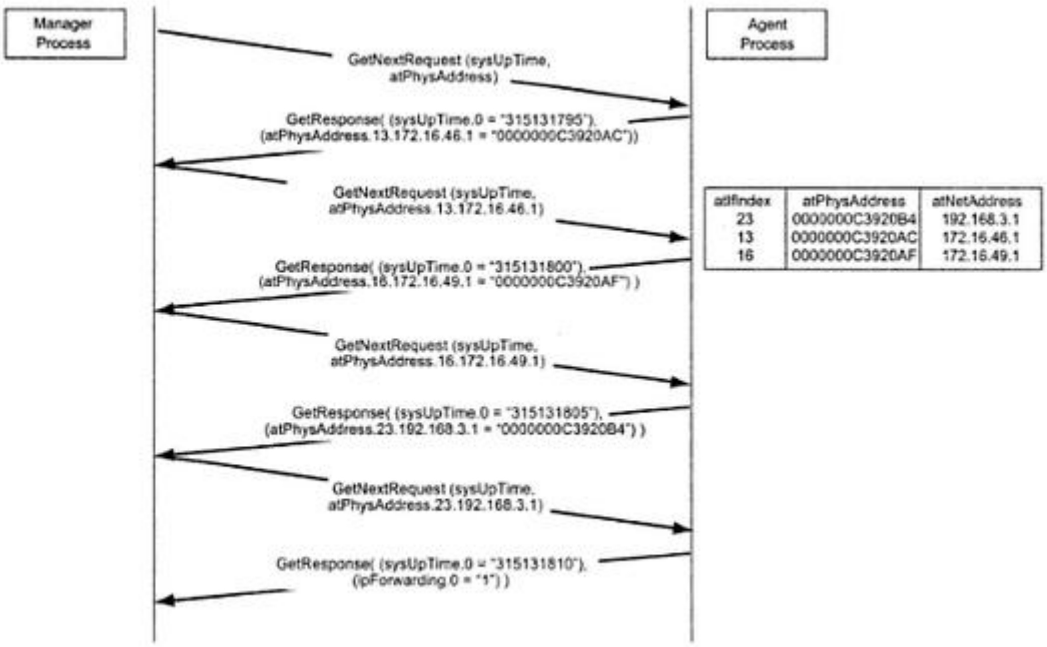


Figure 5.15 Get-Next-Request Operation for a MIB in Figure 5.12

USE OF GETNEXTREQUEST-PDU OPERATION

- In fig:5.12, the first two objects, A and B, are single-valued scalar objects.
- They are followed by an aggregate object represented by the table T with an entry E and two rows of three columnar objects, T.E.1.1 through T.E.3.2.
- The MIB group ends with a scalar object Z.
- fig:5.13 shows the use of nine get-request messages to retrieve the nine objects.


Manager Process
Agent Process
GetNextRequest (sysUpTime, atPhysAddress)
GetResponse((sysUpTime.0 = "315131795"), (atPhysAddress.13.172.16.46.1 = "0000000C3920AC"))
GetNextRequest (sysUpTime, atPhysAddress.13.172.16.46.1)
GetResponse((sysUpTime.0 = "315131800"), (atPhysAddress.16.172.16.49.1 = "0000000C3920AF"))
GetNextRequest (sysUpTime, atPhysAddress.16.172.16.49.1)
GetResponse((sysUpTime.0 = "315131805"), (atPhysAddress.23.192.168.3.1 = "0000000C3920B4"))
GetNextRequest (sysUpTime, atPhysAddress.23.192.168.3.1)
GetResponse((sysUpTime.0 = "315131810"), (ipForwarding.0 = "1"))

atIfIndex	atPhysAddress	atNetAddress
23	0000000C3920B4	192.168.3.1
13	0000000C3920AC	172.16.46.1
16	0000000C3920AF	172.16.49.1

Figure 5.16 GetNextRequest Example with Indices

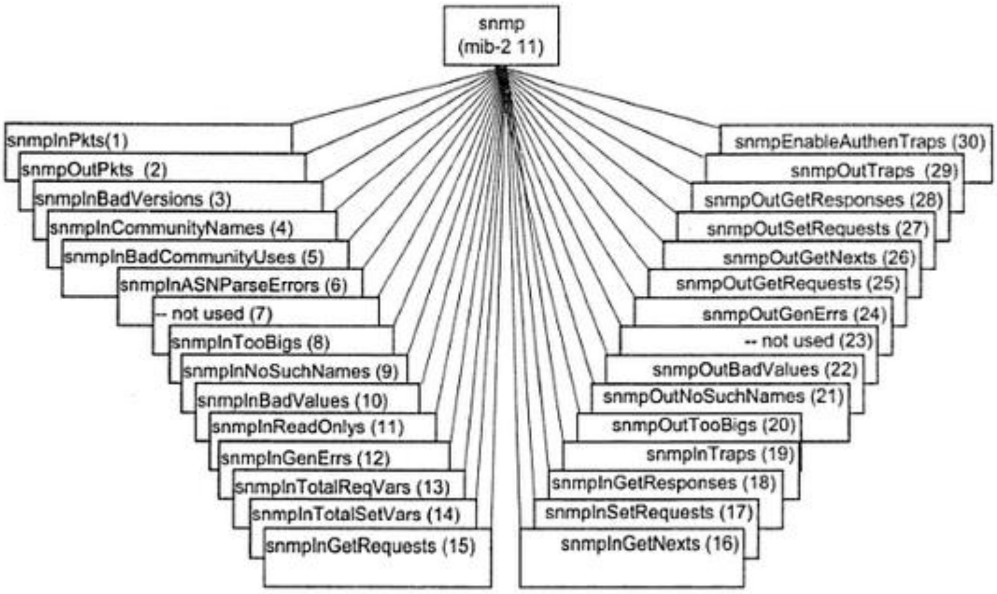
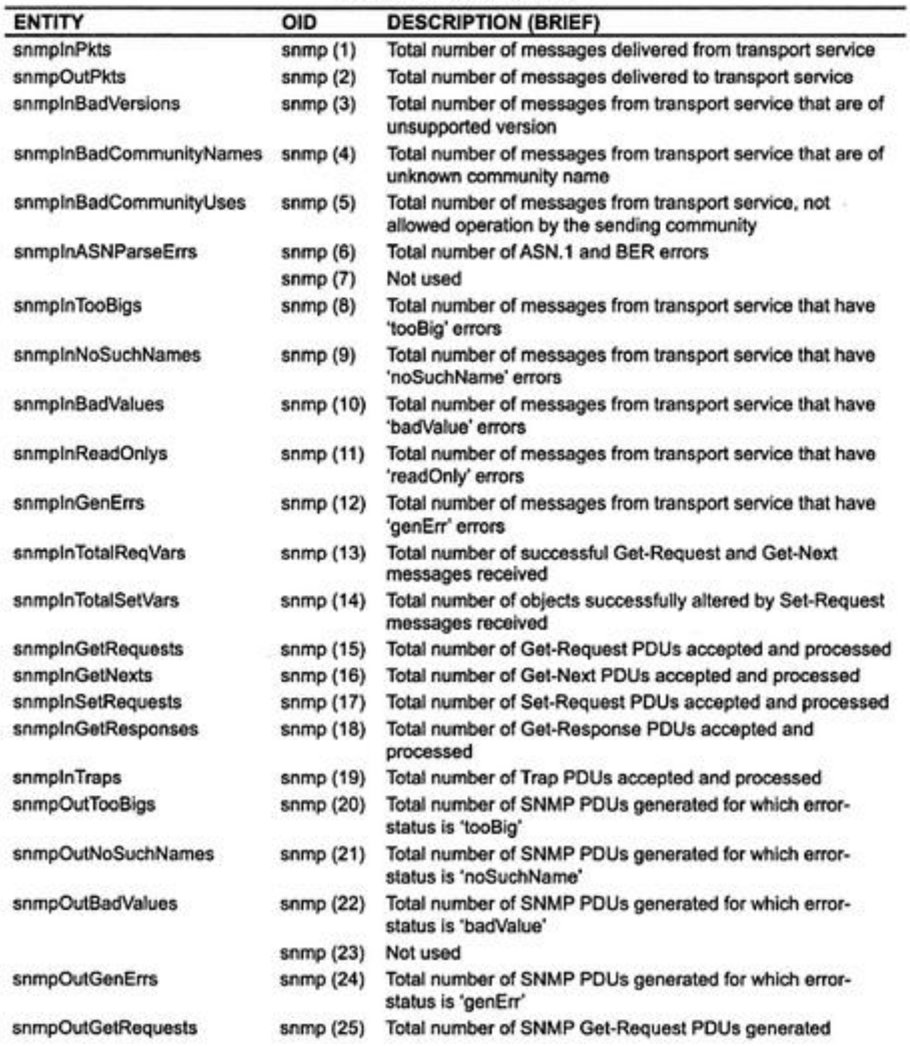


Figure 5.21 SNMP Group

Table 5.4 SNMP Group



ENTITY	OID	DESCRIPTION (BRIEF)
snmpInPkts	snmp (1)	Total number of messages delivered from transport service
snmpOutPkts	snmp (2)	Total number of messages delivered to transport service
snmpInBadVersions	snmp (3)	Total number of messages from transport service that are of unsupported version
snmpInBadCommunityNames	snmp (4)	Total number of messages from transport service that are of unknown community name
snmpInBadCommunityUses	snmp (5)	Total number of messages from transport service, not allowed operation by the sending community
snmpInASNParseErrs	snmp (6)	Total number of ASN.1 and BER errors
	snmp (7)	Not used
snmpInTooBigs	snmp (8)	Total number of messages from transport service that have 'tooBig' errors
snmpInNoSuchNames	snmp (9)	Total number of messages from transport service that have 'noSuchName' errors
snmpInBadValues	snmp (10)	Total number of messages from transport service that have 'badValue' errors
snmpInReadOnlys	snmp (11)	Total number of messages from transport service that have 'readOnly' errors
snmpInGenErrs	snmp (12)	Total number of messages from transport service that have 'genErr' errors
snmpInTotalReqVars	snmp (13)	Total number of successful Get-Request and Get-Next messages received
snmpInTotalSetVars	snmp (14)	Total number of objects successfully altered by Set-Request messages received
snmpInGetRequests	snmp (15)	Total number of Get-Request PDUs accepted and processed
snmpInGetNexts	snmp (16)	Total number of Get-Next PDUs accepted and processed
snmpInSetRequests	snmp (17)	Total number of Set-Request PDUs accepted and processed
snmpInGetResponses	snmp (18)	Total number of Get-Response PDUs accepted and processed
snmpInTraps	snmp (19)	Total number of Trap PDUs accepted and processed
snmpOutTooBigs	snmp (20)	Total number of SNMP PDUs generated for which error-status is 'tooBig'
snmpOutNoSuchNames	snmp (21)	Total number of SNMP PDUs generated for which error-status is 'noSuchName'
snmpOutBadValues	snmp (22)	Total number of SNMP PDUs generated for which error-status is 'badValue'
	snmp (23)	Not used
snmpOutGenErrs	snmp (24)	Total number of SNMP PDUs generated for which error-status is 'genErr'
snmpOutGetRequests	snmp (25)	Total number of SNMP Get-Request PDUs generated

UNIT 5: SNMP MANAGEMENT – RMON

WHAT IS REMOTE MONITORING?

- The monitored information, gathered & analyzed locally, can be transmitted to a remote network management station. In such a case, remotely monitoring the network with a probe is referred to as RMON (Remote Network Monitoring) (fig:8.1).
- Two remote LANs, one a token ring LAN and another, an FDDI LAN ,are connected to the backbone network. The NMS is on the local Ethernet LAN.
- An Ethernet probe is on the Ethernet LAN monitoring the local LAN The FDDI backbone is monitored by an FDDI probe via the bridge and Ethernet LAN A token ring probe monitors the token ring LAN. It communicates with the NMS via the routers ,the WAN & the backbone network The remote FDDI is monitored by the built-in probe on the router. The FDDI probe communicates with NMS.
- All 4 probes that monitor the 4 LANs and communicate with the NMS are RMON devices.

Advantages:

1) Each RMON device monitors the local network segment and does the necessary analyses.

This relays information in both solicited & unsolicited fashion to the NMS.

For example, RMON could be locally polling the network elements in a segment. If it detects an abnormal condition such as heavy packet loss or excessive collisions, it sends an alarm. Because the polling in local, the information is fairly reliable.

The local monitoring and reporting to a remote NMS significantly reduces SNMP traffic in the network.

2) RMON reduces the need for agents in the network to be visible at all times to the NMS.

3) Monitoring packets such as ICMP pings, may get lost in long-distance communication, especially under heavy traffic conditions. Such losses may wrongly be interpreted by the NMS that the managed object is down. RMON pings locally and hence has less chance of losing packets, thus increasing monitoring reliability.

4) The individual segments can be monitored almost continuously. This capability provides better statistics and control.

Thus a fault can be diagnosed more quickly by the RMON and reported to the NMS.

5) RMON provides higher network availability for users and greater productivity for administrators.

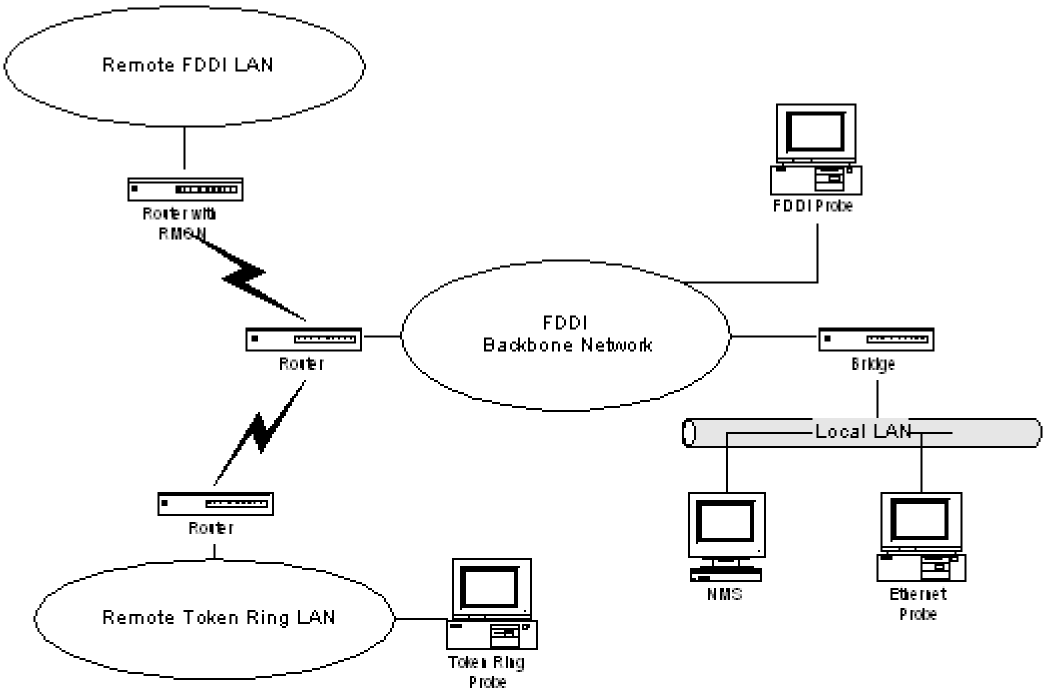


Figure 8.1 Network Configuration with RMONs

RMON1 GROUPS & FUNCTIONS

- The data gathering modules, which are LAN probes, gather data from the remotely monitored network. comprising Ethernet & token ring LANs. The data can serve as inputs to 4 sets of functions, 3 of which monitor traffic statistics(fig:8.3).
- The functions performed by various groups is as follows:
  1) Statistics: provides link level statistics.
  2) History: collects periodic statistical data & stores them for later retrieval.
  3) Alarm: generates events when the data sample gathered crosses pre-established threshold.
  4) Host: gathers statistical data on hosts.
  5) Host Top N: computes the top N hosts on the respective categories of statistics gathered.
  6) Matrix: gathers statistics on traffic between pairs of hosts.
  7) Filter: performs filter function that enables capture of desired parameters.
  8) Packet capture: provides packet capture capability for gathering packets after they flow through a channel.
  9) Event: controls the generation of events & notifications.
- The outputs of the various modules are analyzed & presented in tabular and graphical forms to the user by the network manager in the NMS.
- The filter group is a cascade of 2 filters. The packet filter filters incoming packets by performing a Boolean and/or XOR with a mask specified. The filtered packet stream is considered a channel, and we can make further selections based on the channel mask.
- The filtered outputs may generate either alarms or events, which are reported to the network manager. The output of the filter group can be stored in the packet capture module for further analysis by the network manager.

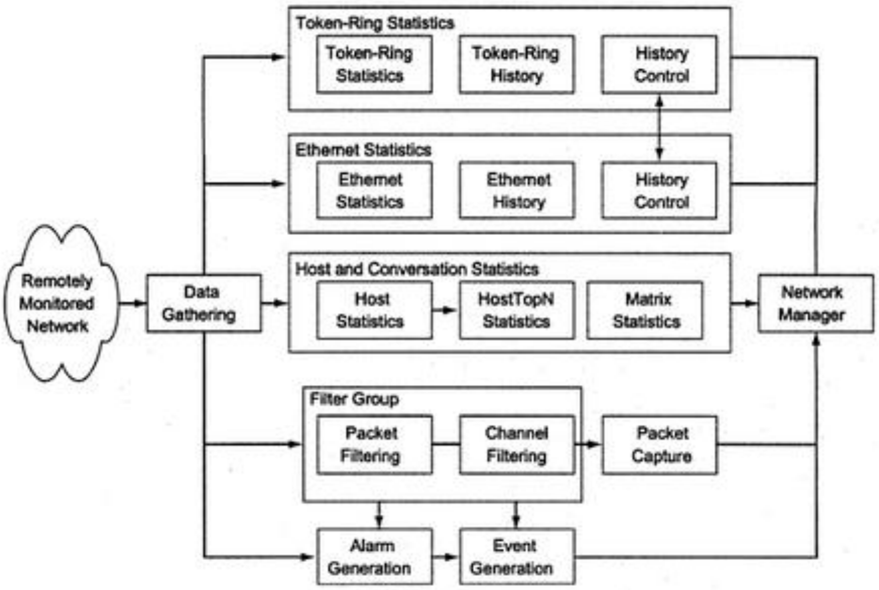


Figure 8.3 RMON1 Groups and Functions

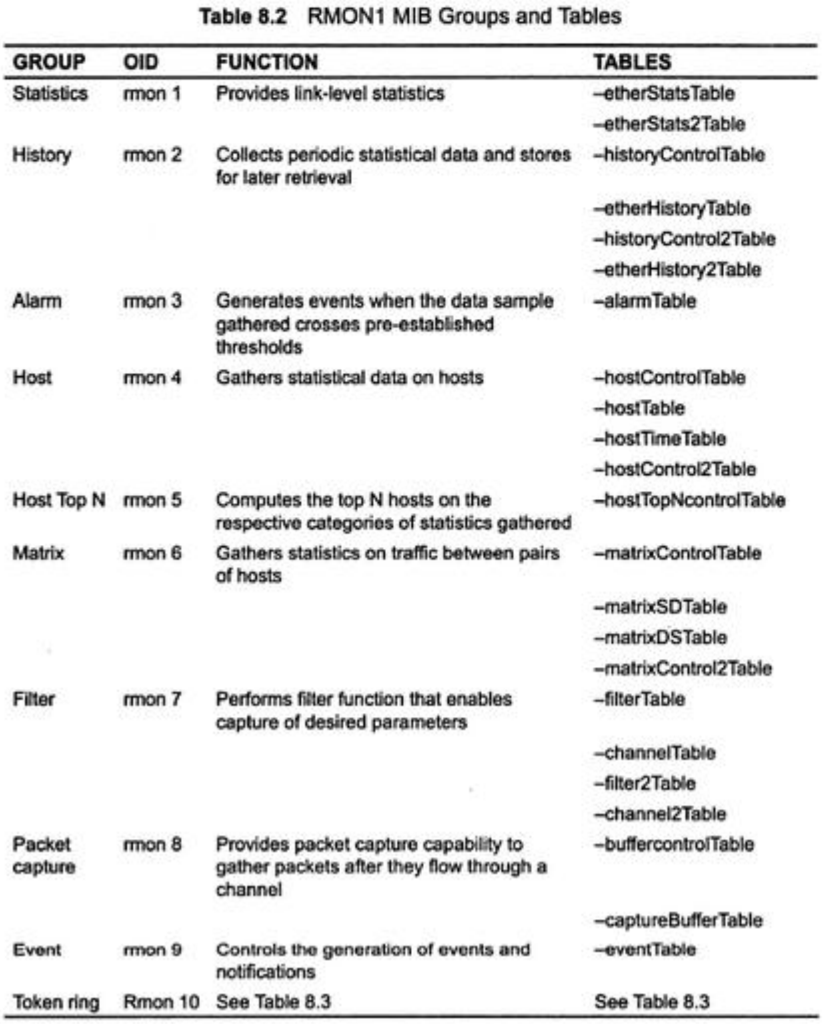


Table 8.2 RMON1 MIB Groups and Tables

GROUP	OID	FUNCTION	TABLES
Statistics	rmon 1	Provides link-level statistics	–etherStatsTable –etherStats2Table
History	rmon 2	Collects periodic statistical data and stores for later retrieval	–historyControlTable –etherHistoryTable –historyControl2Table –etherHistory2Table
Alarm	rmon 3	Generates events when the data sample gathered crosses pre-established thresholds	–alarmTable
Host	rmon 4	Gathers statistical data on hosts	–hostControlTable –hostTable –hostTimeTable –hostControl2Table
Host Top N	rmon 5	Computes the top N hosts on the respective categories of statistics gathered	–hostTopNcontrolTable
Matrix	rmon 6	Gathers statistics on traffic between pairs of hosts	–matrixControlTable –matrixSDTable –matrixDSTable –matrixControl2Table
Filter	rmon 7	Performs filter function that enables capture of desired parameters	–filterTable –channelTable –filter2Table –channel2Table
Packet capture	rmon 8	Provides packet capture capability to gather packets after they flow through a channel	–buffercontrolTable –captureBufferTable
Event	rmon 9	Controls the generation of events and notifications	–eventTable
Token ring	Rmon 10	See Table 8.3	See Table 8.3

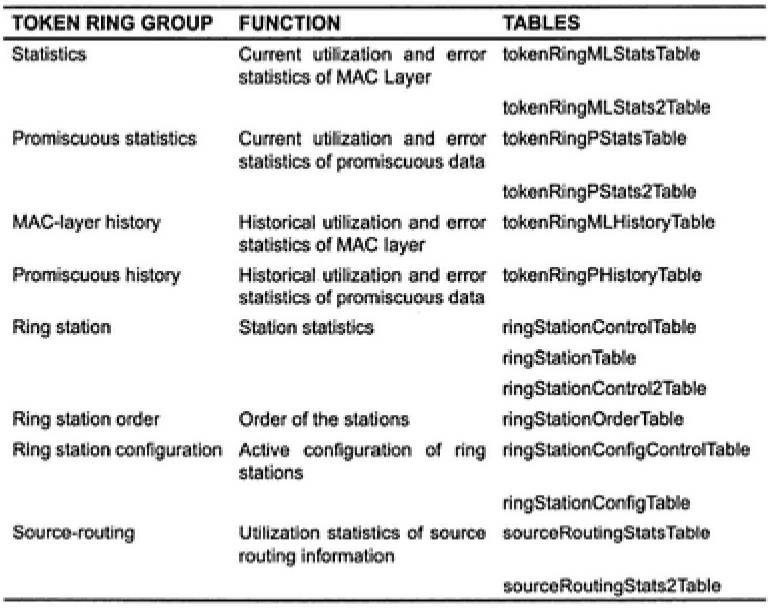
THE RMON MANAGEMENT INFORMATION BASE

- The RMON2 MIB is arranged in 10 groups:

  1) The protocol directory group identifies the protocols that the probe can monitor.
  2) The protocol distribution group provides information on the relative traffic of different protocols either in octet or packets. It collects basic statistics that help a NMS manage bandwidth allocation utilized by different protocols.
  3) The address map group binds the MAC address to network address on each interface.
  4) The network layer host group measures the traffic sent from and to each network address representing each host discovered by the probe.
  5) The network layer matrix group provides information on the conversation between pairs of hosts in both directions.
  6) Both application layer host and application layer matrix groups calculate traffic by protocol units and use their respective control tables in the network layer host group and the network layer matrix group.
  7) Alarm and history information are combined into the user history collection group. This function, normally done by NMS, can be off-loaded to RMON.
  8) The probe configuration group provides the facility for configuring the probe.The data can be accessed via a modem connection.

RMON TOKEN RING MIB GROUPS & TABLES

Table 8.3 RMON Token-Ring MIB Groups and Tables



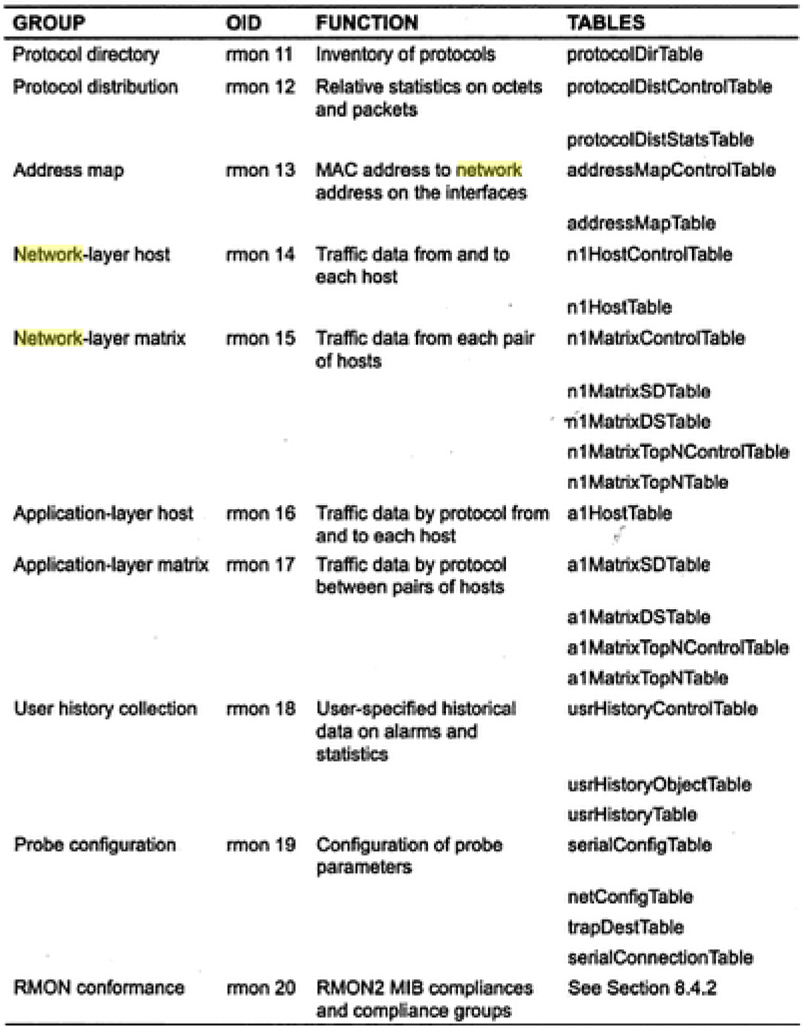
TOKEN RING GROUP	FUNCTION	TABLES
Statistics	Current utilization and error statistics of MAC Layer	tokenRingMLStatsTable tokenRingMLStats2Table
Promiscuous statistics	Current utilization and error statistics of promiscuous data	tokenRingPStatsTable tokenRingPStats2Table
MAC-layer history	Historical utilization and error statistics of MAC layer	tokenRingMLHistoryTable
Promiscuous history	Historical utilization and error statistics of promiscuous data	tokenRingPHistoryTable
Ring station	Station statistics	ringStationControlTable ringStationTable ringStationControl2Table
Ring station order	Order of the stations	ringStationOrderTable
Ring station configuration	Active configuration of ring stations	ringStationConfigControlTable ringStationConfigTable
Source-routing	Utilization statistics of source routing information	sourceRoutingStatsTable sourceRoutingStats2Table

1) The MAC layer statistics group collects data on token ring parameters such token packets ,errors in packets ,bursts ,polling etc.
2) The promiscuous statistics group collects statistics on the number of packets of various sizes and the type of packets-- multicast or broadcast data.
3) The ring station group provides statistics on each station being monitored on the ring ,along with its status. The data are stored in the ringStationTable. The rings and parameters to be monitored are controlled by the ringStationControlTable.
4) The ring station order group provides the order of the station on the monitored rings & has only a data table
5) The ring station configuration group manages the stations on the ring.
6) The Source routing group gather statistics on routing information in a pure source routing environment.

RMON2 MIB GROUPS AND TABLES

- Applicable to Layers 3 and above.
- Functions similar to RMON1.
- Enhancement to RMON1.
- Defined conformance and compliance.

Table 8.4 RMON2 MIB Groups and Tables



GROUP	OID	FUNCTION	TABLES
Protocol directory	rmon 11	Inventory of protocols	protocolDirTable
Protocol distribution	rmon 12	Relative statistics on octets and packets	protocolDistControlTable protocolDistStatsTable
Address map	rmon 13	MAC address to network address on the interfaces	addressMapControlTable addressMapTable
Network-layer host	rmon 14	Traffic data from and to each host	n1HostControlTable n1HostTable
Network-layer matrix	rmon 15	Traffic data from each pair of hosts	n1MatrixControlTable n1MatrixSDTable n1MatrixDSTable n1MatrixTopNControlTable n1MatrixTopNTable
Application-layer host	rmon 16	Traffic data by protocol from and to each host	a1HostTable
Application-layer matrix	rmon 17	Traffic data by protocol between pairs of hosts	a1MatrixSDTable a1MatrixDSTable a1MatrixTopNControlTable a1MatrixTopNTable
User history collection	rmon 18	User-specified historical data on alarms and statistics	usrHistoryControlTable usrHistoryObjectTable usrHistoryTable
Probe configuration	rmon 19	Configuration of probe parameters	serialConfigTable netConfigTable trapDestTable serialConnectionTable
RMON conformance	rmon 20	RMON2 MIB compliances and compliance groups	See Section 8.4.2

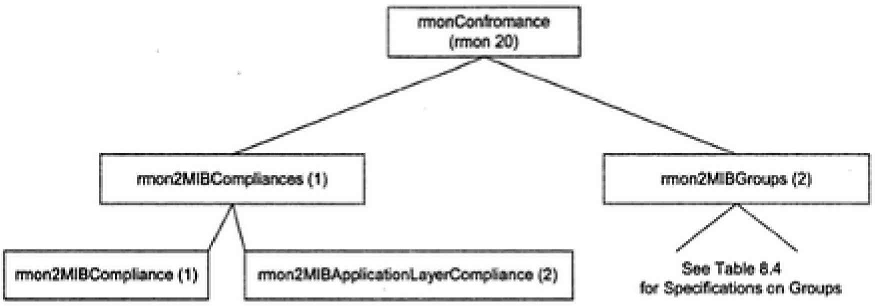

rmonConfromance (rmon 20)
rmon2MIBCompliances (1)
rmon2MIBGroups (2)
rmon2MIBCompliance (1)
rmon2MIBApplicationLayerCompliance (2)
See Table 8.4 for Specifications on Groups

Figure 8.6 RMON2 Conformance Group

ATM REMOTE MONITORING

- Switch extensions for RMON & ATM RMON define RMON objects at the base layer, which is the ATM .sublayer. ATM protocol IDs for RMON2 define additional objects needed at the higher levels. (Fig:8.7) .
- Extending RMON to ATM devices requires design changes and new functionality.
- Particular attention must be paid to high-speed requirements, cells versus frames, and the connection-oriented nature of ATM.
- The high-speed nature of ATM imposes a severe set of requirements in ATM RMON implementation.
- At the data link sublayer, ATM RMON measures cells instead of packets or frames, and provides cell-based per-host and per-conversation traffic statistics At the application layer, RMON provides basic statistics for each monitored cell stream, for each ATM host, and for conversations between pair-wise hosts .
- It also provides the capability for flexible configuration mechanisms suited to the connection-oriented nature of ATM.
- When RMON instrumentation is embedded in the switch fabric (part c & d) ,no modification of the circuit is needed. In part a & b , circuit steering is needed to copy the cells onto the probe(Fig:8.8) .

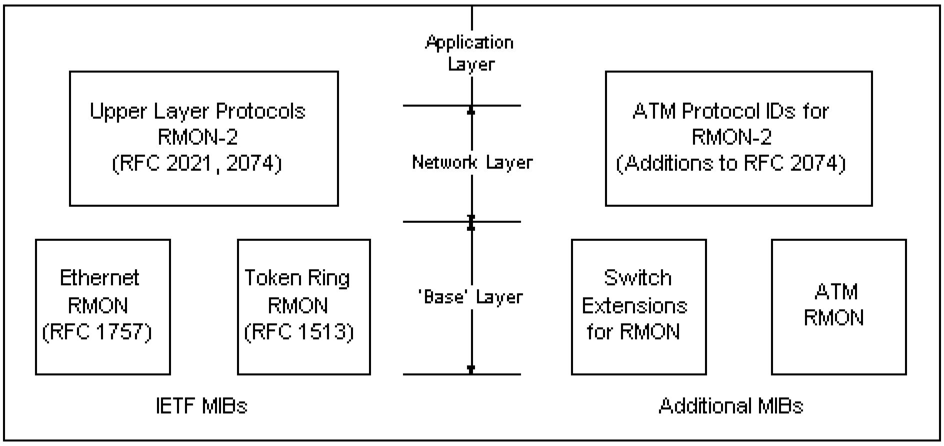

Application Layer
Upper Layer Protocols RMON-2 (RFC 2021, 2074)
Network Layer
ATM Protocol IDs for RMON-2 (Additions to RFC 2074)
Ethernet RMON (RFC 1757)
Token Ring RMON (RFC 1513)
'Base' Layer
Switch Extensions for RMON
ATM RMON
IETF MIBs
Additional MIBs

Figure 8.7 RMON MIB Framework (©1995 ATM Forum)

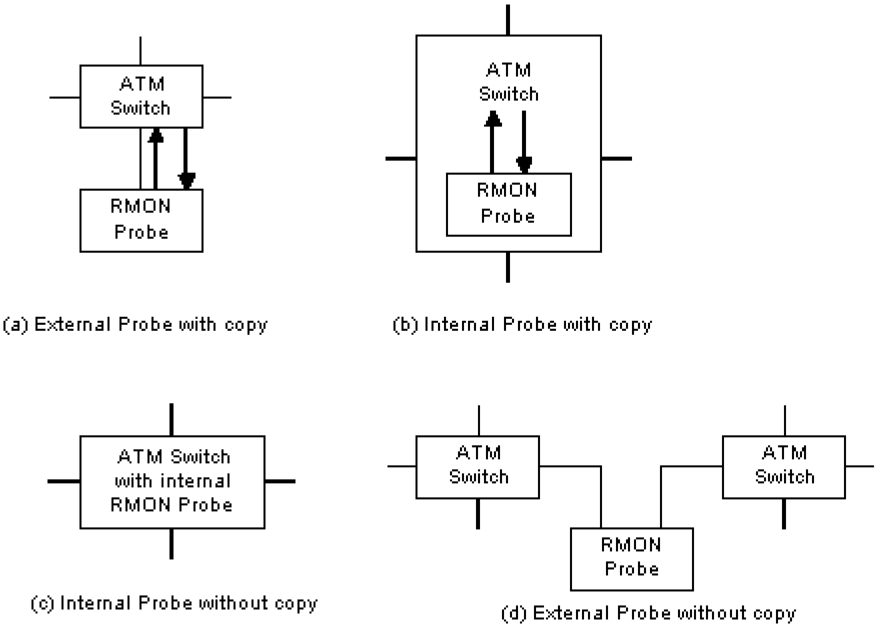

ATM Switch
RMON Probe
(a) External Probe with copy
ATM Switch
RMON Probe
(b) Internal Probe with copy
ATM Switch with internal RMON Probe
(c) Internal Probe without copy
ATM Switch
ATM Switch
RMON Probe
(d) External Probe without copy

Figure 8.8 ATM Probe Location ©1995 ATM Forum)

ATM RMON MIB GROUPS AND TABLES

- ATM RMON MIB contains four groups.
- portSelect group selects ports.
- atmStats collects basic statistics based on portselection.
- atmHost gathers statistics based on host traffic.
- atmMatrix group collects conversation traffic and ranks the top-N entries.

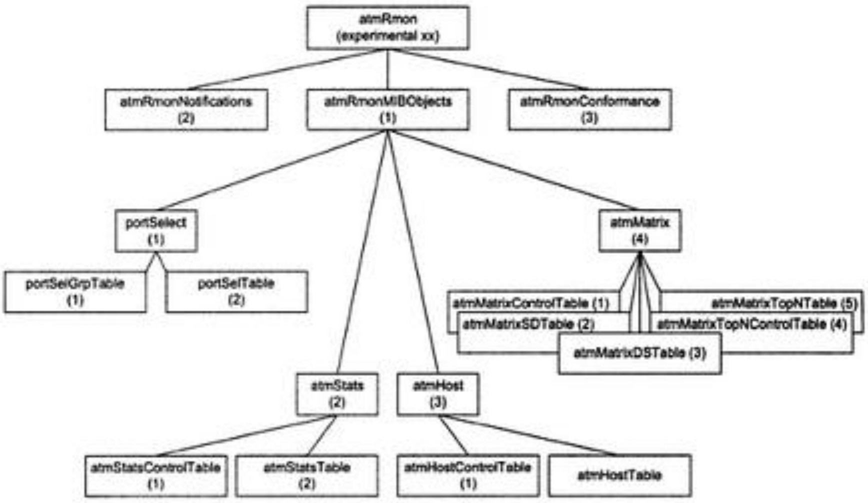
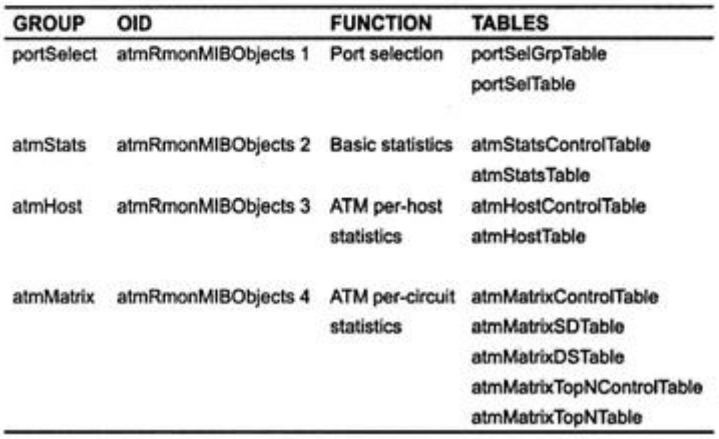


Figure 8.9 ATM RMON MIB

Table 8.6 ATM RMON MIB Groups and Tables



GROUP	OID	FUNCTION	TABLES
portSelect	atmRmonMIBObjects 1	Port selection	portSelGrpTable portSelTable
atmStats	atmRmonMIBObjects 2	Basic statistics	atmStatsControlTable atmStatsTable
atmHost	atmRmonMIBObjects 3	ATM per-host statistics	atmHostControlTable atmHostTable
atmMatrix	atmRmonMIBObjects 4	ATM per-circuit statistics	atmMatrixControlTable atmMatrixSDTable atmMatrixDSTable atmMatrixTopNControlTable atmMatrixTopNTable