Übung MIB und ASN.1

Aufgabenbeschreibung:
Erstellen Sie in Gruppenarbeit eine MIB in ASN.1-Syntax für ein beliebiges Gerät unter folgenden Bedingungen:

- Das Gerät ist kein Hub, Switch, Router, Access-Point, Firewall oder normales Endgerät wie PC/Laptop → Lassen Sie sich etwas einfallen!
- Die MIB muss mit einem MIB-Browser (Internet bzw. von mir geliefert) interpretierbar sein.
- Die MIB enthält mindestens 3 Zweige, davon einer als Sequence.

Erstellen Sie eine kurze Präsentation dazu für Overhead/Beamer/…, die Sie in max. 10 Minuten dem Plenum vorstellen.

Zeitumfang der Übung: 60 Minuten

Hinweise:
- Die folgenden Seiten zeigen Ihnen den Anfang der MIB-II und damit Syntax und Struktur.
- Jede MIB muss in die Gesamtstruktur aller MIBs passen. Für solche Übungen, Tests etc. wurde der Zweig „Experimental“ geschaffen.
- Beim Test mit dem MIB-Browser steht Ihnen ja nicht das reale Gerät zur Verfügung, entsprechend können Sie natürlich auch keinen Wert auslesen oder setzen. Die Navigation durch die Baumstruktur sollte auch so funktionieren.
Beispiel: MIB II gemäß RFC 1213, erste Seiten

RFC1213-MIB DEFINITIONS ::= BEGIN
IMPORTS
  mgmt, NetworkAddress, IpAddress, Counter, Gauge, TimeTicks
  FROM RFC1155-SMI
OBJECT-TYPE
  FROM RFC-1212;

mgmt OBJECT IDENTIFIER ::= { iso org(3) dod(6) internet(1) mgmt(2) }
directory OBJECT IDENTIFIER ::= { internet 1 }
experimental OBJECT IDENTIFIER ::= { internet 3 }
private OBJECT IDENTIFIER ::= { internet 4 }
Enterprises OBJECT IDENTIFIER ::= { private 1 }

-- This MIB module uses the extended OBJECT-TYPE macro as
-- defined in [14];

-- MIB-II (same prefix as MIB-I)
mib-2 OBJECT IDENTIFIER ::= { mgmt 1 }

-- textual conventions

-- DisplayString ::= 
--    OCTET STRING
-- This data type is used to model textual information taken from the NVT
-- ASCII character set. By convention, objects with this syntax are declared
-- as having
--
-- SIZE (0..255)

-- PhysAddress ::= 
--    OCTET STRING
-- This data type is used to model media addresses. For many types of media,
-- this will be in a binary representation.
-- For example, an ethernet address would be represented as a string of
-- 6 octets.

-- groups in MIB-II
system OBJECT IDENTIFIER ::= { mib-2 1 }
interfaces OBJECT IDENTIFIER ::= { mib-2 2 }
at OBJECT IDENTIFIER ::= { mib-2 3 }
ip OBJECT IDENTIFIER ::= { mib-2 4 }
icmp OBJECT IDENTIFIER ::= { mib-2 5 }
tcp OBJECT IDENTIFIER ::= { mib-2 6 }
udp OBJECT IDENTIFIER ::= { mib-2 7 }
egp OBJECT IDENTIFIER ::= { mib-2 8 }
-- historical (some say hysterical)
-- cmot     OBJECT IDENTIFIER ::= { mib-2 9 }

transmission OBJECT IDENTIFIER ::= { mib-2 10 }

snmp     OBJECT IDENTIFIER ::= { mib-2 11 }

-- the System group
-- Implementation of the System group is mandatory for all
-- systems. If an agent is not configured to have a value
-- for any of these variables, a string of length 0 is returned.

sysDescr OBJECT-TYPE
   SYNTAX DisplayString (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 only contain
       printable ASCII characters."
   ::= { system 1 }

sysObjectID OBJECT-TYPE
   SYNTAX   OBJECT IDENTIFIER
   ACCESS   read-only
   STATUS   mandatory
   DESCRIPTION
       "The vendor's authoritative identification of the
       network management subsystem contained in the
       entity. This value is allocated within the SMI
       enterprises subtree (1.3.6.1.4.1) and provides an
       easy and unambiguous means for determining `what
       kind of box' is being managed. For example, if
       vendor 'Flintstones, Inc.' was assigned the
       subtree 1.3.6.1.4.1.4242, it could assign the
       identifier 1.3.6.1.4.1.4242.1.1 to its 'Fred Router'."
   ::= { system 2 }

sysUpTime OBJECT-TYPE
   SYNTAX   TimeTicks
   ACCESS   read-only
   STATUS   mandatory
   DESCRIPTION
       "The time (in hundredths of a second) since the
       network management portion of the system was last
       re-initialized."
   ::= { system 3 }
sysContact OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..255))
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
"The textual identification of the contact person for this managed node, together with information on how to contact this person."
 ::= { system 4 }

sysName OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..255))
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
"An administratively-assigned name for this managed node. By convention, this is the node's fully-qualified domain name."
 ::= { system 5 }

sysLocation OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..255))
ACCESS  read-write
STATUS  mandatory
DESCRIPTION
"The physical location of this node (e.g. `telephone closet, 3rd floor')."
 ::= { system 6 }

sysServices OBJECT-TYPE
SYNTAX  INTEGER (0..127)
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
"A value which indicates the set of services that this entity primarily offers. The value is a sum. This sum initially takes the value zero. Then, for each layer, L, in the range 1 through 7, that this node performs transactions for, 2 raised to (L - 1) is added to the sum. For example, a node which performs primarily routing functions would have a value of 4 (2^(3-1)). In contrast, a node which is a host offering application services would have a value of 72 (2^(4-1) + 2^(7-1)). Note that in the context of the Internet suite of protocols, values should be calculated accordingly:

layer   functionality
1   physical (e.g., repeaters)
2   datalink/subnetwork (e.g., bridges)
3   internet (e.g., IP gateways)
4   end-to-end (e.g., IP hosts)
7   applications (e.g., mail relays)"
For systems including OSI protocols, layers 5 and 6 may also be counted.

\[\text{::= \{ system 7 \} }\]

-- the Interfaces group

-- Implementation of the Interfaces group is mandatory for all systems

ifNumber OBJECT-TYPE
SYNTAX  INTEGER
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
 "The number of network interfaces (regardless of their current state) present on this system."
 ::= { interfaces 1 }

-- the Interfaces table

-- The Interfaces table contains information on the entity's interfaces. Each interface is thought of as being attached to a `subnetwork'. Note that this term should not be confused with `subnet' which refers to an addressing partitioning scheme used in the Internet suite of protocols.

ifTable OBJECT-TYPE
SYNTAX  SEQUENCE OF IfEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION
 "A list of interface entries. The number of entries is given by the value of ifNumber."
 ::= { interfaces 2 }

ifEntry OBJECT-TYPE
SYNTAX  IfEntry
ACCESS  not-accessible
STATUS  mandatory
DESCRIPTION
 "An interface entry containing objects at the subnetwork layer and below for a particular interface."
INDEX  { ifIndex }
 ::= { ifTable 1 }
IfEntry ::=  
SEQUENCE {  
     ifIndex    INTEGER,  
     ifDescr    DisplayString,  
     ifType     INTEGER,  
     ifMtu      INTEGER,  
     ifSpeed    Gauge,  
     ifPhysAddress    PhysAddress,  
     ifAdminStatus    INTEGER,  
     ifOperStatus    INTEGER,  
     ifLastChange    TimeTicks,  
     ifInOctets      Counter,  
     ifInUcastPkts   Counter,  
     ifInNUcastPkts  Counter,  
     ifInDiscards   Counter,  
     ifInErrors     Counter,  
     ifInUnknownProtos     Counter,  
     ifOutOctets      Counter,  
     ifOutUcastPkts   Counter,  
     ifOutNUcastPkts  Counter,  
     ifOutDiscards   Counter,  
     ifOutErrors     Counter,  
     ifOutQLen       Gauge,  
     ifSpecific    OBJECT IDENTIFIER  
}
ifIndex OBJECT-TYPE
   SYNTAX  INTEGER
   ACCESS  read-only
   STATUS  mandatory
   DESCRIPTION
      "A unique value for each interface. Its value ranges between 1 and
      the value of ifNumber. The value for each interface must remain
      constant at least from one re-initialization of the entity's
      network management system to the next re-initialization."
   ::= { ifEntry 1 }

ifDescr OBJECT-TYPE
   SYNTAX  DisplayString (SIZE (0..255))
   ACCESS  read-only
   STATUS  mandatory
   DESCRIPTION
      "A textual string containing information about the interface.
      This string should include the name of the manufacturer,
      the product name and the version of the hardware interface."
   ::= { ifEntry 2 }

ifType OBJECT-TYPE
   SYNTAX  INTEGER {
      other(1),          -- none of the following
      regular1822(2),
      hdh1822(3),
      ddn-x25(4),
      rfc877-x25(5),
      ethernet-csmacd(6),
      iso88023-csmacd(7),
      iso88024-tokenBus(8),
      iso88025-tokenRing(9),
      iso88026-man(10),
      starLan(11),
      proteon-10Mbit(12),
      proteon-80Mbit(13),
      hyperchannel(14),
      fddi(15),
      lapb(16),
      sdlc(17),
      ds1(18),          -- T-1
      e1(19),           -- European equiv. of T-1
      basicISDN(20),    -- Proprietary serial
      primaryISDN(21),  -- Proprietary serial
      propPointToSerial(22),
      ppp(23),
      softwareLoopback(24),
      eon(25),          -- CLNP over IP [11]
      ethernet-3Mbit(26),
      nsip(27),         -- XNS over IP
      slip(28),         -- Generic SLIP
ultra(29),          -- ULTRA technologies
    ds3(30),            -- T-3
    sip(31),            -- SMDS
    frame-relay(32)

    } ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
    "The type of interface, distinguished according to the physical/link
    protocol(s) immediately 'below' the network layer in the protocol stack."
    ::= { ifEntry 3 }

ifMtu OBJECT-TYPE
SYNTAX  INTEGER
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
"The size of the largest datagram which can be sent/received on the
interface, specified in octets. For interfaces that are used for
transmitting network datagrams, this is the size of the largest
network datagram that can be sent on the interface."
 ::= { ifEntry 4 }

ifSpeed OBJECT-TYPE
SYNTAX  Gauge
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
"An estimate of the interface's current bandwidth in bits per second.
For interfaces which do not vary in bandwidth or for those where no
accurate estimation can be made, this object should contain the
nominal bandwidth."
 ::= { ifEntry 5 }

[...]