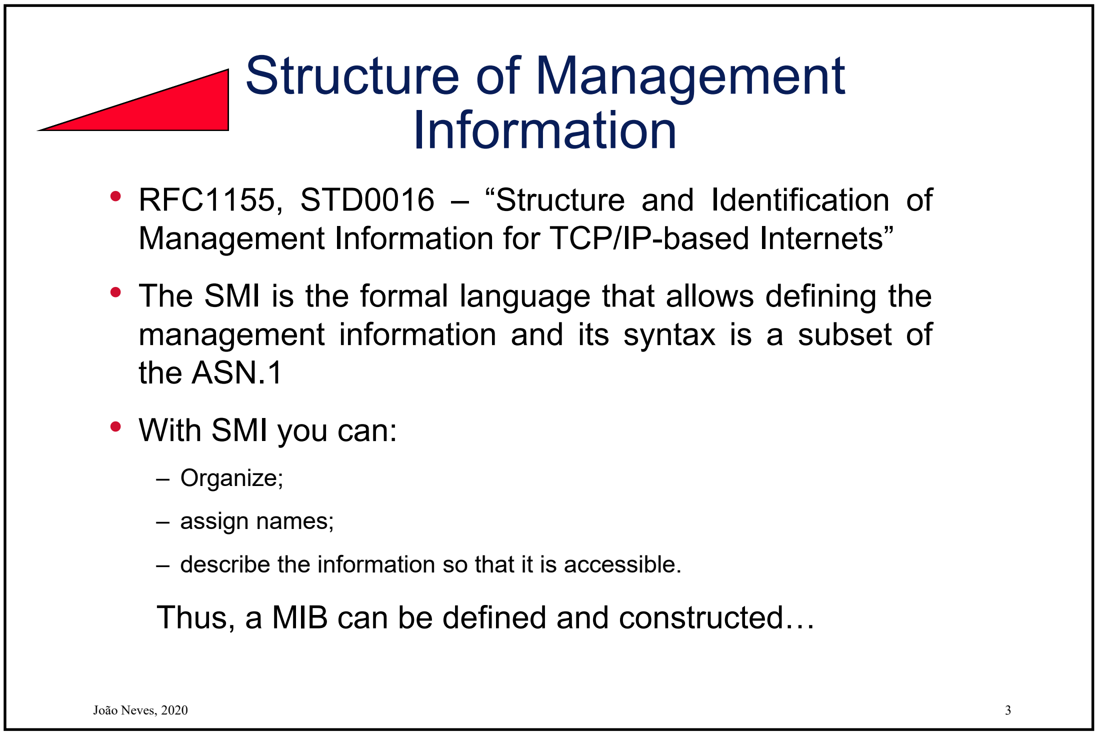
# SMI & MIB

The Management Information

## IETF Internet Management

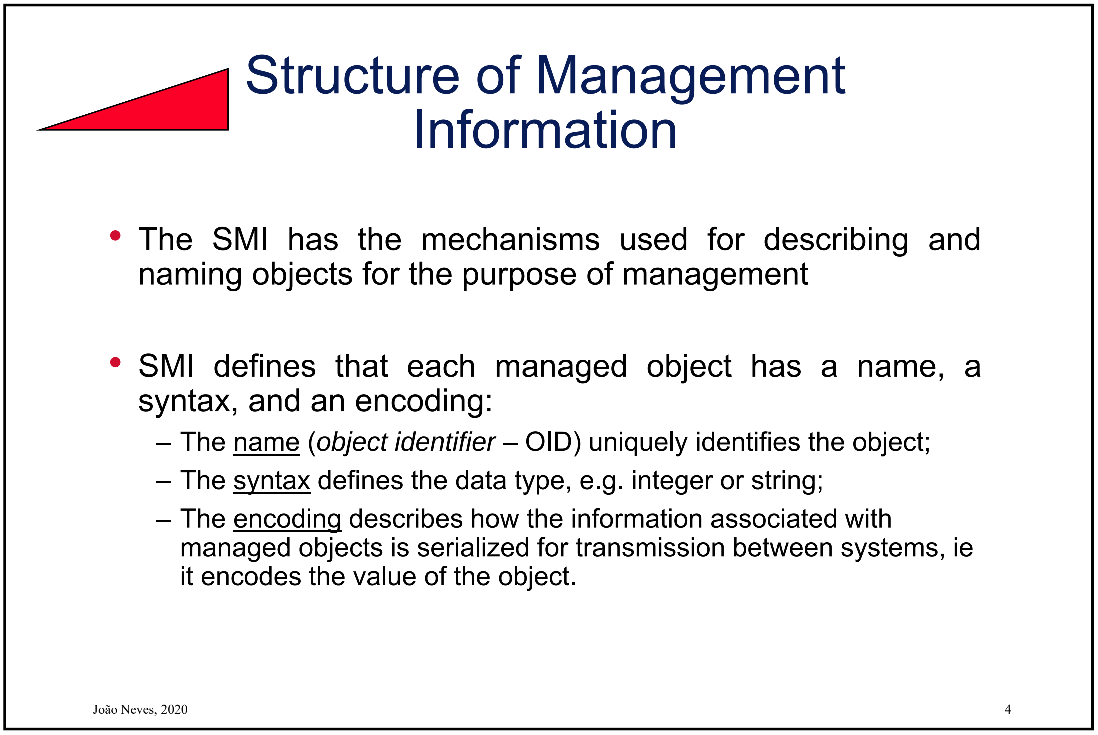
- Based on the Simple Network Management Protocol, but is more than a protocol, is a complete framework:
  - **A data definition language** - The Structure and Identification of Management Information (SMI)
  - **Definitions of management information** - Instrumentation described in the Management Information Base (MIB)
  - **Protocol definition** - The Simple Network Management Protocol (SNMP)



# Structure of Management Information

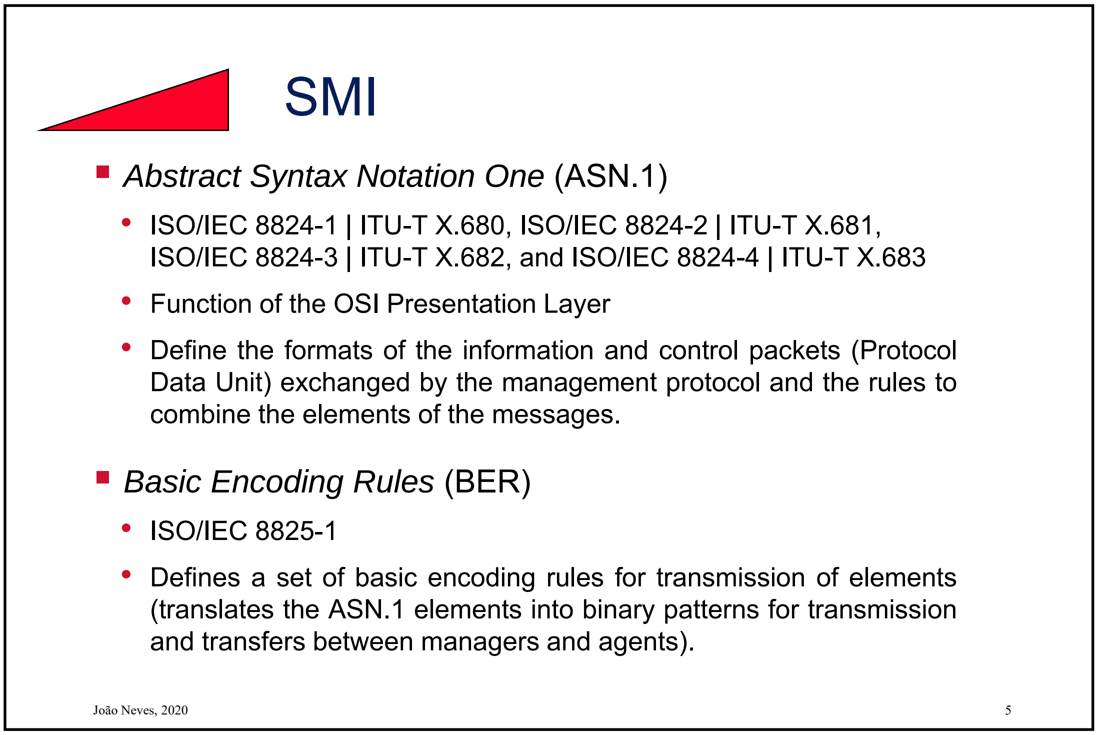
- RFC1155, STD0016 – “Structure and Identification of Management Information for TCP/IP-based Internets”
- The SMI is the formal language that allows defining the management information and its syntax is a subset of the ASN.1
- With SMI you can:
  - Organize;
  - assign names;
  - describe the information so that it is accessible.

Thus, a MIB can be defined and constructed…



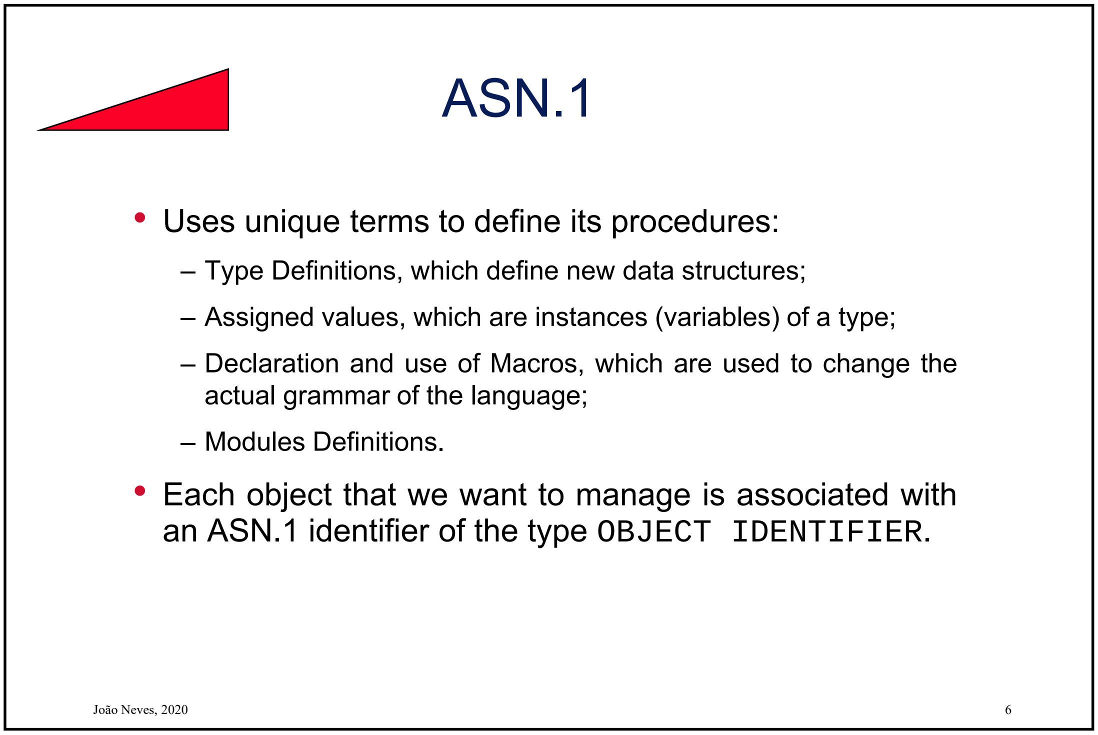
# Structure of Management Information

- The SMI has the mechanisms used for describing and naming objects for the purpose of management
- SMI defines that each managed object has a name, a syntax, and an encoding:
  - The name (*object identifier* – OID) uniquely identifies the object;
  - The syntax defines the data type, e.g. integer or string;
  - The encoding describes how the information associated with managed objects is serialized for transmission between systems, ie it encodes the value of the object.



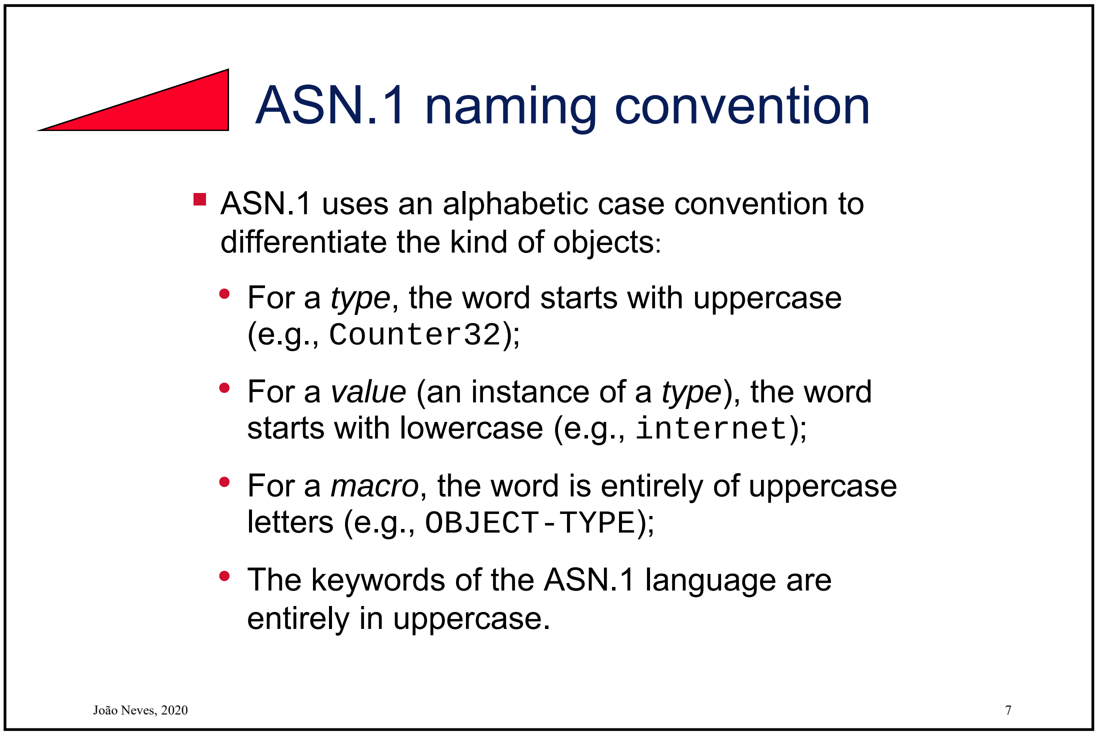
## SMI

- *Abstract Syntax Notation One* (ASN.1)
  - ISO/IEC 8824-1 | ITU-T X.680, ISO/IEC 8824-2 | ITU-T X.681, ISO/IEC 8824-3 | ITU-T X.682, and ISO/IEC 8824-4 | ITU-T X.683
  - Function of the OSI Presentation Layer
  - Define the formats of the information and control packets (Protocol Data Unit) exchanged by the management protocol and the rules to combine the elements of the messages.
- *Basic Encoding Rules* (BER)
  - ISO/IEC 8825-1
  - Defines a set of basic encoding rules for transmission of elements (translates the ASN.1 elements into binary patterns for transmission and transfers between managers and agents).



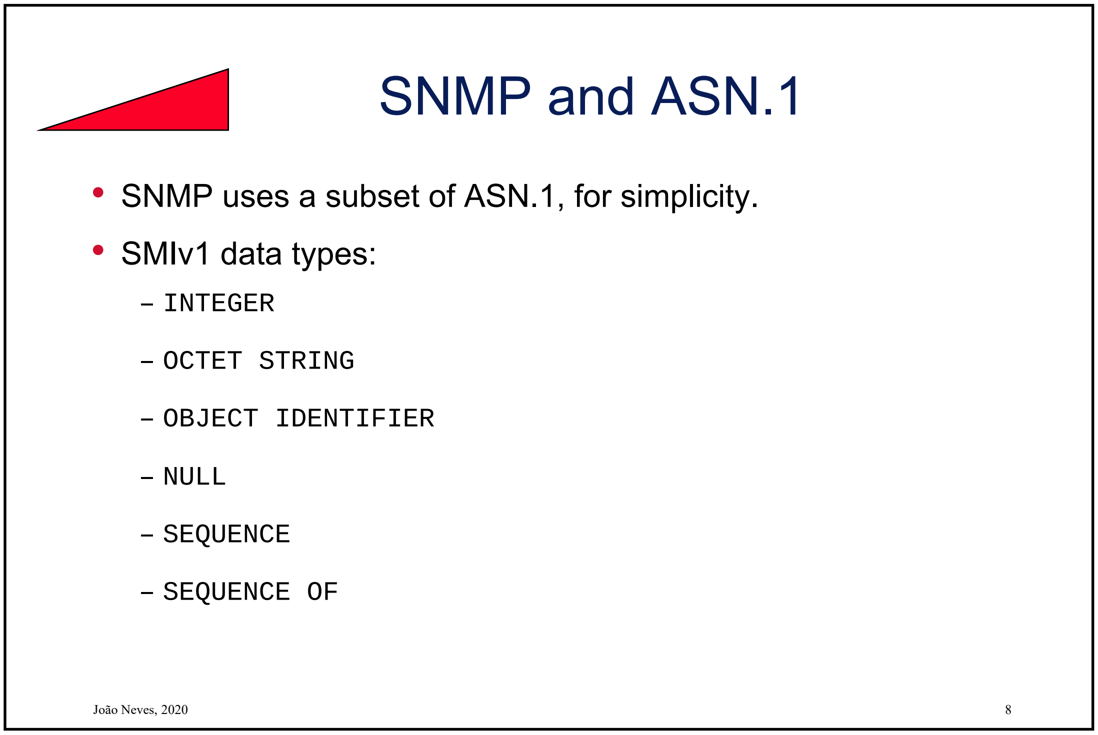
## ASN.1

- Uses unique terms to define its procedures:
  - – Type Definitions, which define new data structures;
  - – Assigned values, which are instances (variables) of a type;
  - – Declaration and use of Macros, which are used to change the actual grammar of the language;
  - – Modules Definitions.
- Each object that we want to manage is associated with an ASN.1 identifier of the type `OBJECT IDENTIFIER`.



# ASN.1 naming convention

- ASN.1 uses an alphabetic case convention to differentiate the kind of objects:
  - For a *type*, the word starts with uppercase (e.g., `Counter32`);
  - For a *value* (an instance of a *type*), the word starts with lowercase (e.g., `internet`);
  - For a *macro*, the word is entirely of uppercase letters (e.g., `OBJECT-TYPE`);
  - The keywords of the ASN.1 language are entirely in uppercase.



# SNMP and ASN.1

- SNMP uses a subset of ASN.1, for simplicity.
- SMIv1 data types:
  - `INTEGER`
  - `OCTET STRING`
  - `OBJECT IDENTIFIER`
  - `NULL`
  - `SEQUENCE`
  - `SEQUENCE OF`

# Types

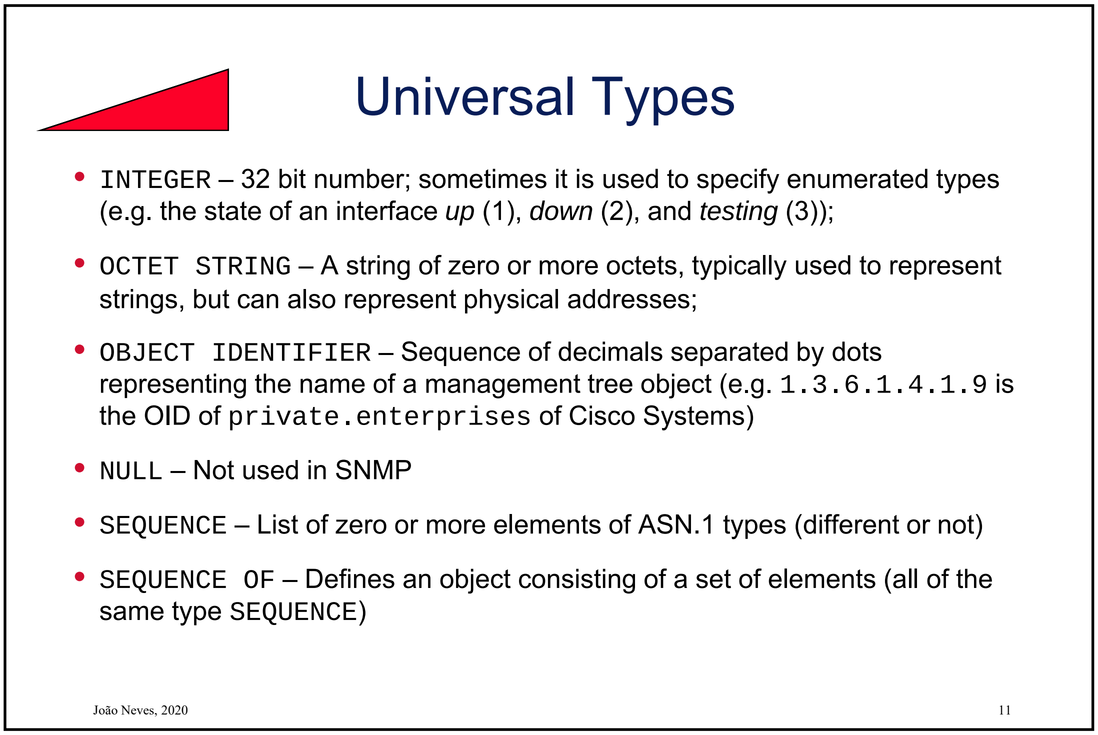
- The *type* classifies data (`SYNTAX` statement):
  - *Primitive Types* – `INTEGER, OCTET STRING, OBJECT IDENTIFIER, NULL` (all in uppercase);
  - *Constructed Types* – generates lists and tables (`SEQUENCE` and `SEQUENCE OF`);
  - *Defined Types* – alternative names defined in SMI for simple or complex ASN.1 types (usually more descriptive, e.g. `TimeTicks`, `Counter`, `Gauge` or `IpAdress` which represents a 32-bit IP address).

# Types and Values

- The Value quantifies the Type;
- Sometimes only a limited set of values is allowed

  An integer value limited to 8 bits would be

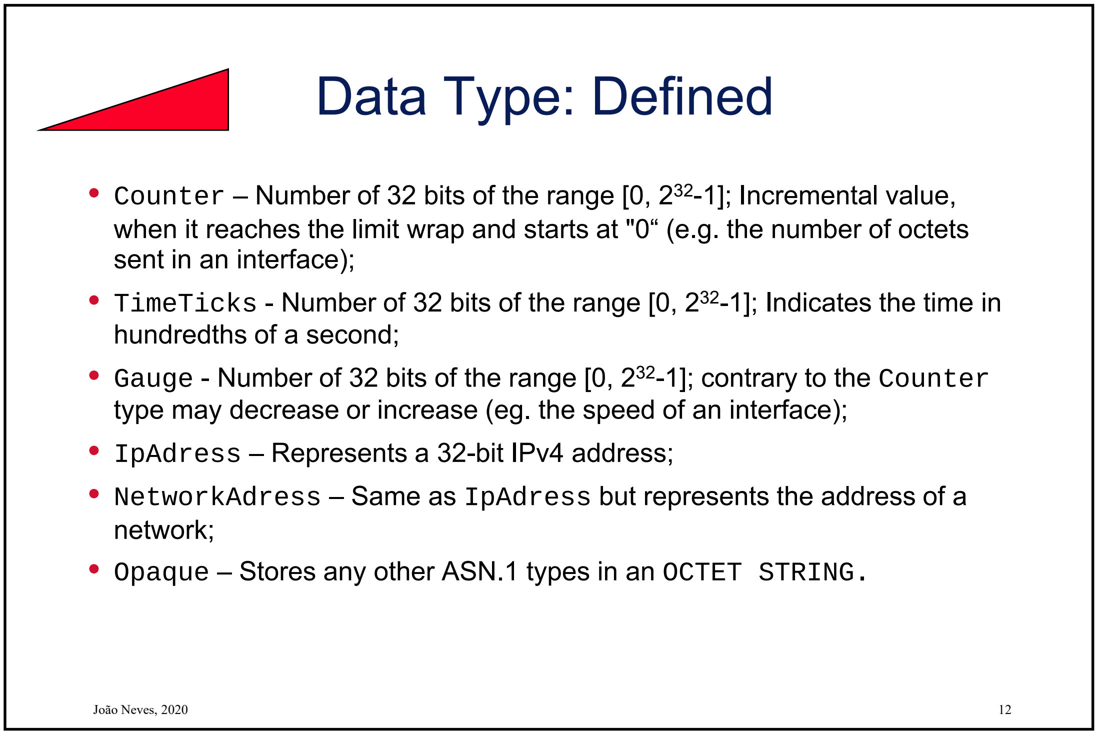
  `INTEGER (0..255)`



# Universal Types

- `INTEGER` – 32 bit number; sometimes it is used to specify enumerated types (e.g. the state of an interface *up* (1), *down* (2), and *testing* (3));
- `OCTET STRING` – A string of zero or more octets, typically used to represent strings, but can also represent physical addresses;
- `OBJECT IDENTIFIER` – Sequence of decimals separated by dots representing the name of a management tree object (e.g. `1.3.6.1.4.1.9` is the OID of `private.enterprises` of Cisco Systems)
- `NULL` – Not used in SNMP
- `SEQUENCE` – List of zero or more elements of ASN.1 types (different or not)
- `SEQUENCE OF` – Defines an object consisting of a set of elements (all of the same type `SEQUENCE`)

João Neves, 2020 11



# Data Type: Defined

- `Counter` – Number of 32 bits of the range [0, $2^{32}$-1]; Incremental value, when it reaches the limit wrap and starts at "0“ (e.g. the number of octets sent in an interface);
- `TimeTicks` - Number of 32 bits of the range [0, $2^{32}$-1]; Indicates the time in hundredths of a second;
- `Gauge` - Number of 32 bits of the range [0, $2^{32}$-1]; contrary to the `Counter` type may decrease or increase (eg. the speed of an interface);
- `IpAdress` – Represents a 32-bit IPv4 address;
- `NetworkAdress` – Same as `IpAdress` but represents the address of a network;
- `Opaque` – Stores any other ASN.1 types in an `OCTET STRING.`

João Neves, 2020 12

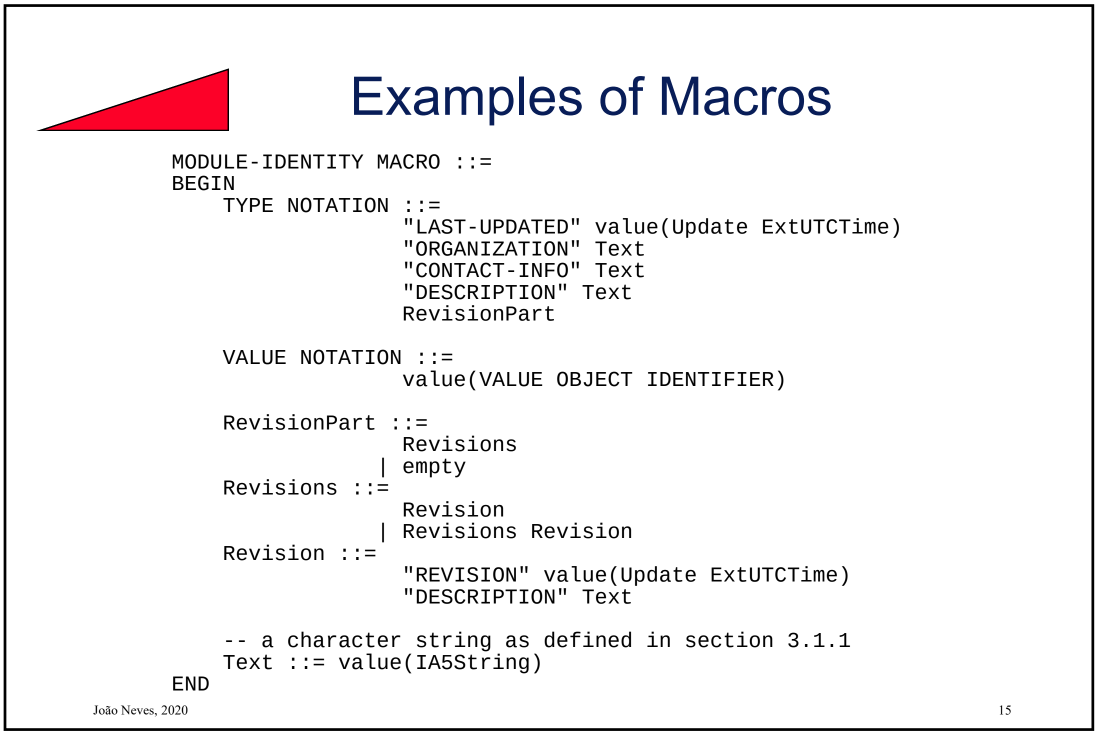
## Macro

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

- Used to define a managed object
- The `MACRO` statement allows the language extension

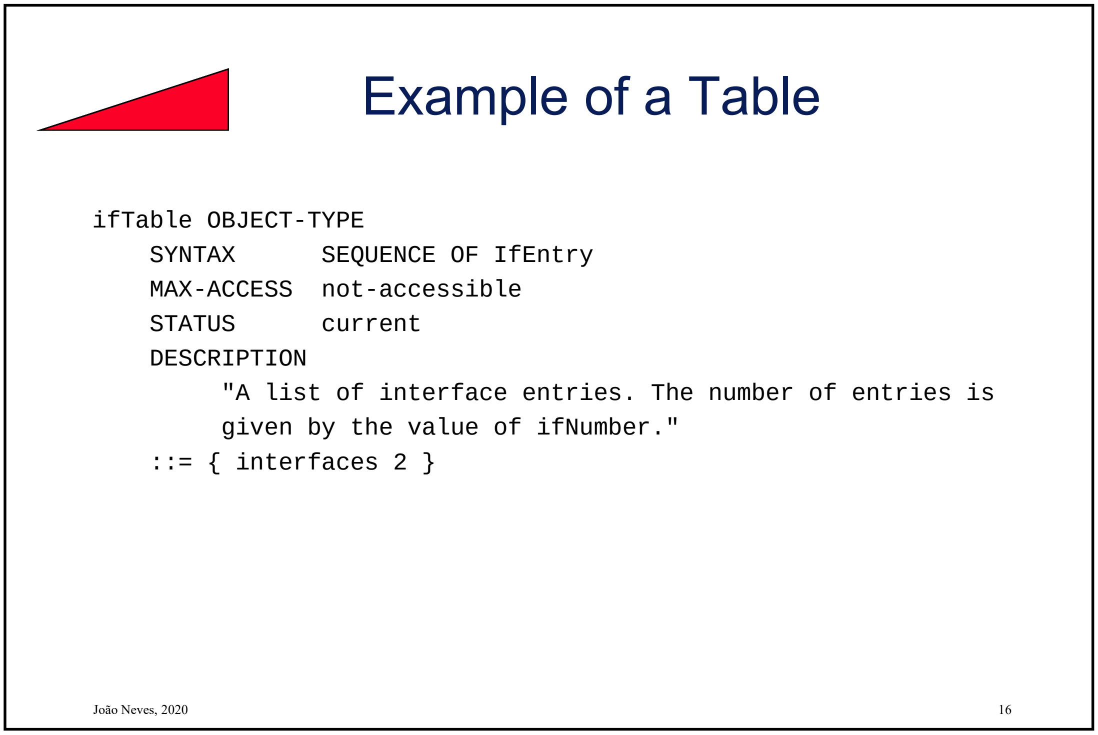
## Declaration of an Object

```
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 }
```



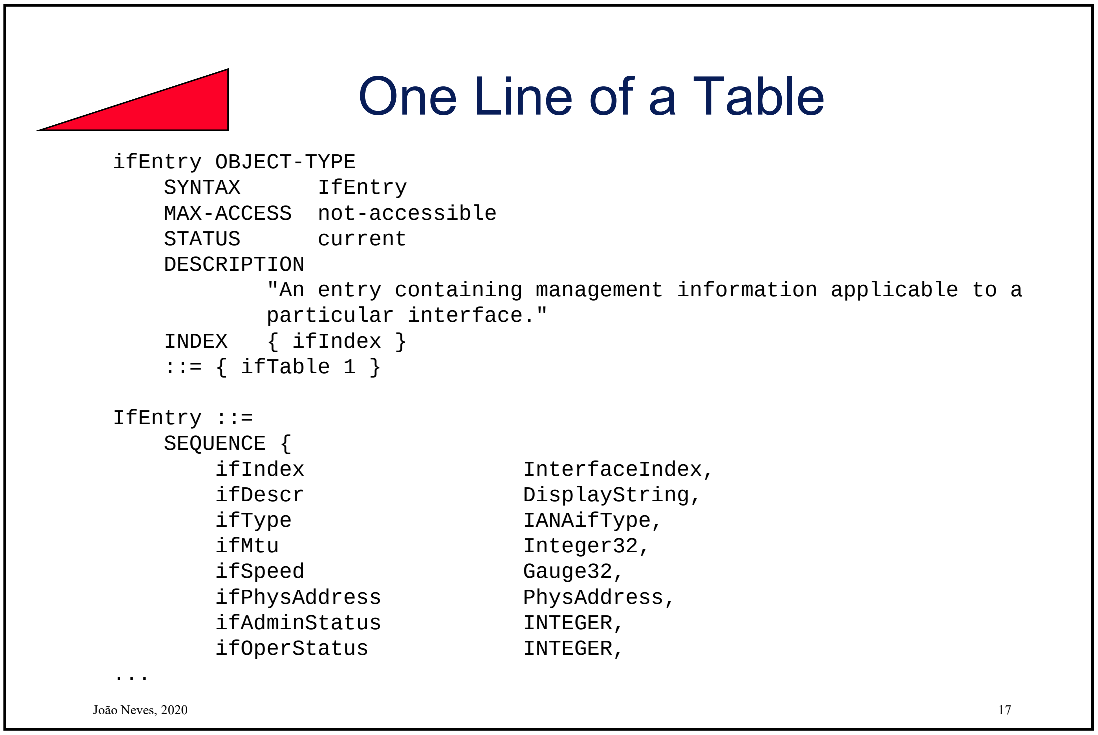
## Examples of Macros

```
MODULE-IDENTITY MACRO ::=
BEGIN
    TYPE NOTATION ::=
                  "LAST-UPDATED" value(Update ExtUTCTime)
                  "ORGANIZATION" Text
                  "CONTACT-INFO" Text
                  "DESCRIPTION" Text
                  RevisionPart

    VALUE NOTATION ::=
                  value(VALUE OBJECT IDENTIFIER)

    RevisionPart ::=
                  Revisions
                | empty
    Revisions ::=
                  Revision
                | Revisions Revision
    Revision ::=
                  "REVISION" value(Update ExtUTCTime)
                  "DESCRIPTION" Text

    -- a character string as defined in section 3.1.1
    Text ::= value(IA5String)
END
```



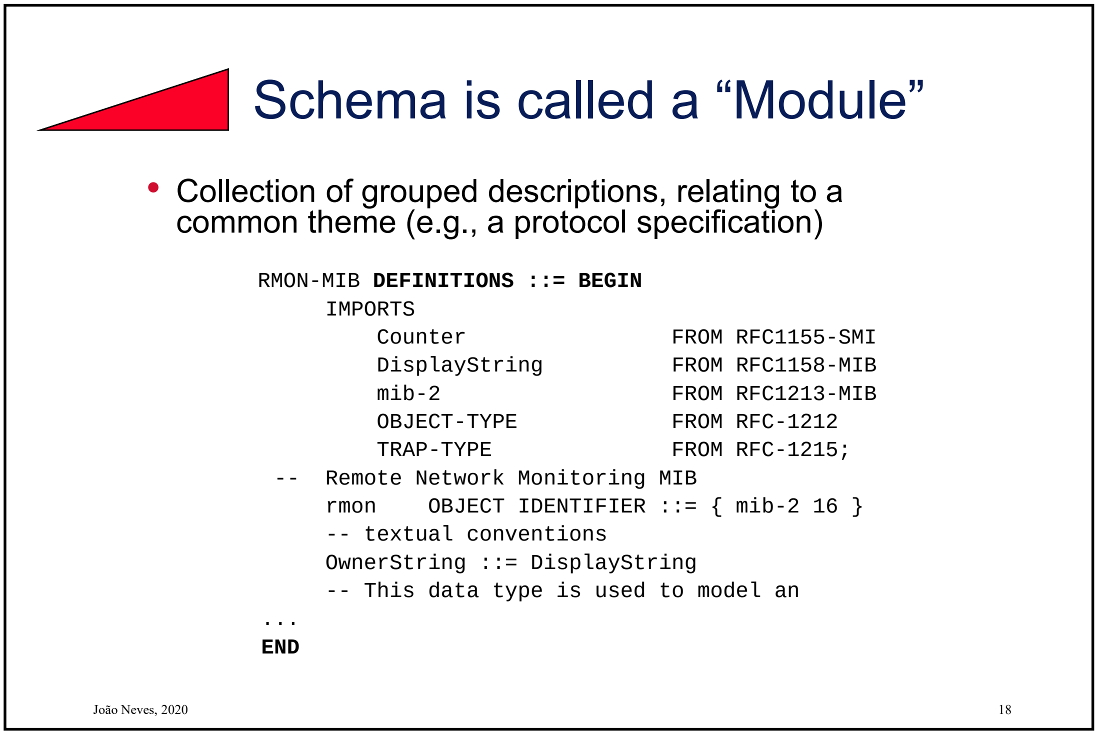
## Example of a Table

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



## One Line of a Table

```
ifEntry OBJECT-TYPE
    SYNTAX      IfEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "An entry containing management information applicable to a
            particular interface."
    INDEX   { ifIndex }
    ::= { ifTable 1 }

IfEntry ::=
    SEQUENCE {
        ifIndex                 InterfaceIndex,
        ifDescr                 DisplayString,
        ifType                  IANAifType,
        ifMtu                   Integer32,
        ifSpeed                 Gauge32,
        ifPhysAddress           PhysAddress,
        ifAdminStatus           INTEGER,
        ifOperStatus            INTEGER,
...
```



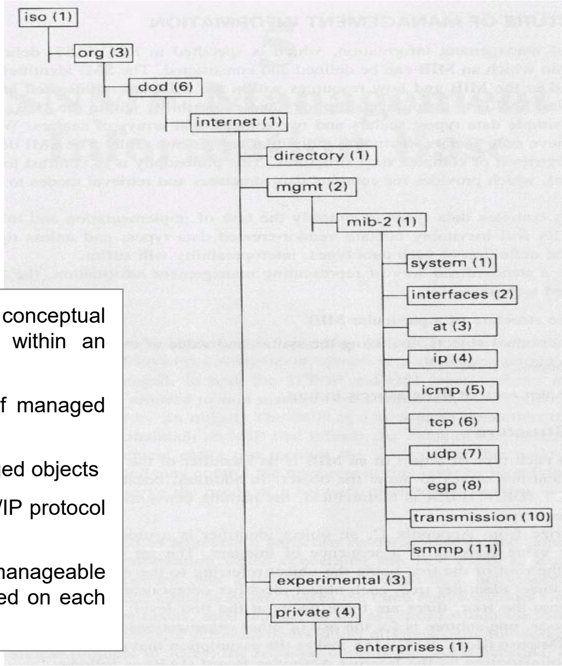
## Schema is called a “Module”

- Collection of grouped descriptions, relating to a common theme (e.g., a protocol specification)

```
RMON-MIB DEFINITIONS ::= BEGIN
     IMPORTS
         Counter                FROM RFC1155-SMI
         DisplayString          FROM RFC1158-MIB
         mib-2                  FROM RFC1213-MIB
         OBJECT-TYPE            FROM RFC-1212
         TRAP-TYPE              FROM RFC-1215;
 --  Remote Network Monitoring MIB
     rmon    OBJECT IDENTIFIER ::= { mib-2 16 }
     -- textual conventions
     OwnerString ::= DisplayString
     -- This data type is used to model an
...
END
```

# Management Information Base

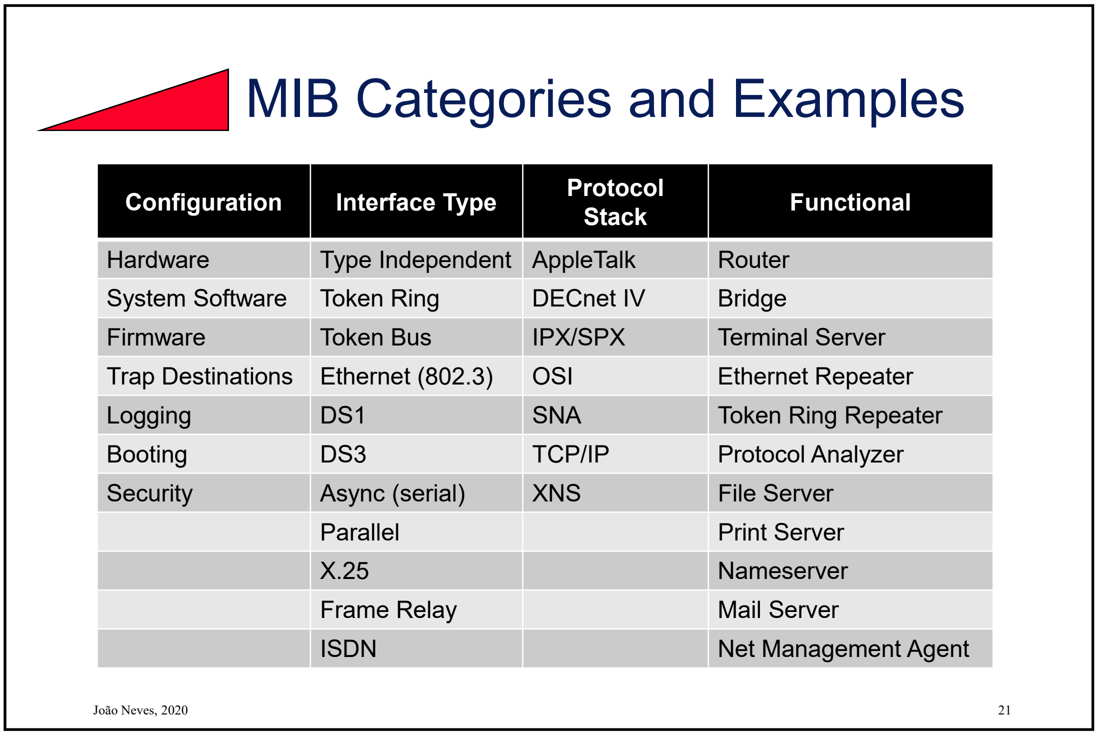
- **Management Information Base**: The conceptual repository of management information within an open system (cf. ISO/IEC 7498-4)
- The MIB specification is the core set of managed objects for the Internet suite of protocols
- Virtual information store to access managed objects
- Define the variables to manage the TCP/IP protocol suite
- Specifies the data elements that a manageable system must have, the operations allowed on each and what the meaning



# MIB

- MIB – RFC 1212, MIB-II – RFC 1213
- The MIB-II is only an extension of the first version and, as such, maintains the same OID

```
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 }
```

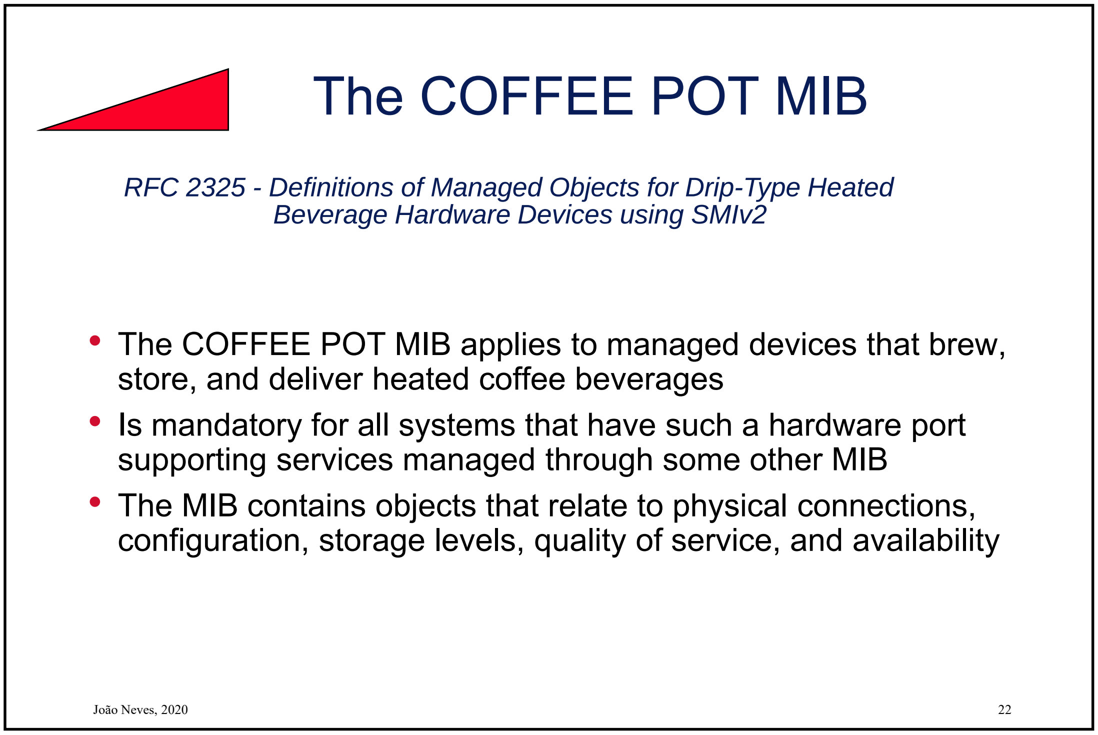


## MIB Categories and Examples

| Configuration | Interface Type | Protocol Stack | Functional |
|---|---|---|---|
| Hardware | Type Independent | AppleTalk | Router |
| System Software | Token Ring | DECnet IV | Bridge |
| Firmware | Token Bus | IPX/SPX | Terminal Server |
| Trap Destinations | Ethernet (802.3) | OSI | Ethernet Repeater |
| Logging | DS1 | SNA | Token Ring Repeater |
| Booting | DS3 | TCP/IP | Protocol Analyzer |
| Security | Async (serial) | XNS | File Server |
|  | Parallel |  | Print Server |
|  | X.25 |  | Nameserver |
|  | Frame Relay |  | Mail Server |
|  | ISDN |  | Net Management Agent |

João Neves, 2020

21



## The COFFEE POT MIB

*RFC 2325 - Definitions of Managed Objects for Drip-Type Heated Beverage Hardware Devices using SMIv2*

- The COFFEE POT MIB applies to managed devices that brew, store, and deliver heated coffee beverages
- Is mandatory for all systems that have such a hardware port supporting services managed through some other MIB
- The MIB contains objects that relate to physical connections, configuration, storage levels, quality of service, and availability

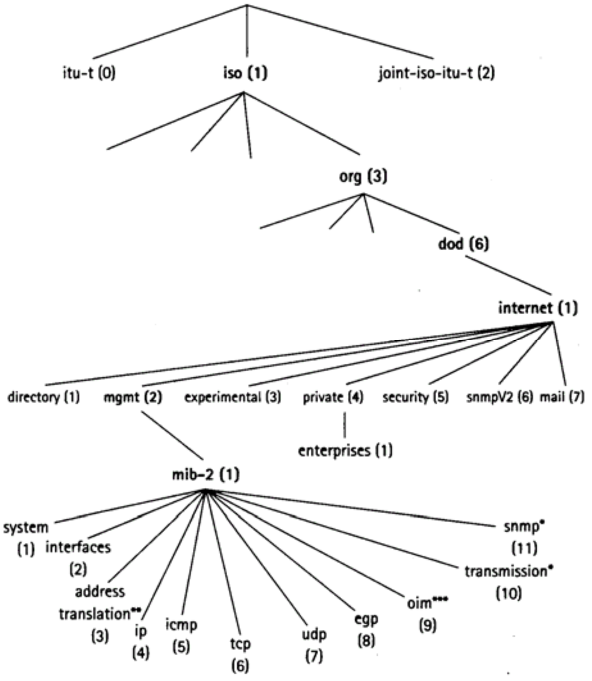
João Neves, 2020

22

## COFFEE-POT-MIB

```
coffee MODULE-IDENTITY
  LAST-UPDATED "9803231700Z"
  ORGANIZATION "Networked Appliance Management Working Group"
  CONTACT-INFO "   Michael Slavitch
                   Loran Technologies,
                   955 Green Valley Crescent
                   […]
               "
  DESCRIPTION
      "The MIB Module for coffee vending devices."
  ::= { transmission 132 }
```

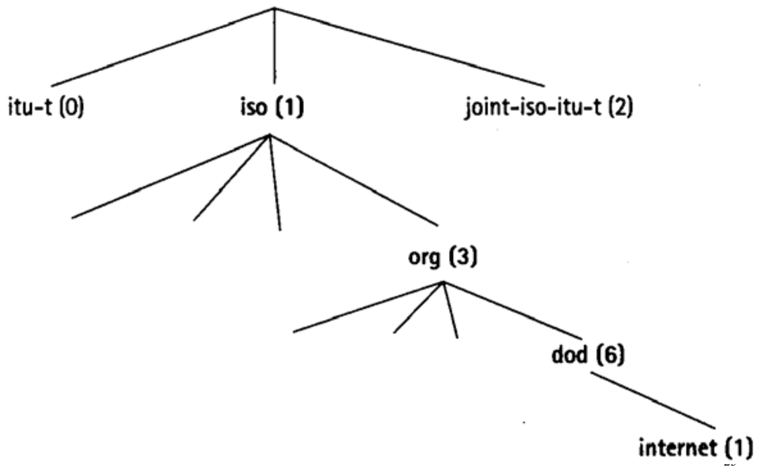
## Name Tree



* Added by MIB-II
** Deprecated by MIB-II
*** Defined in RFC1214

## Names: OID of internet...

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



## The Hierarchy of Names

The root of the subtree for the Internet, administered by the Internet Assigned Numbers Authority (IANA), is:

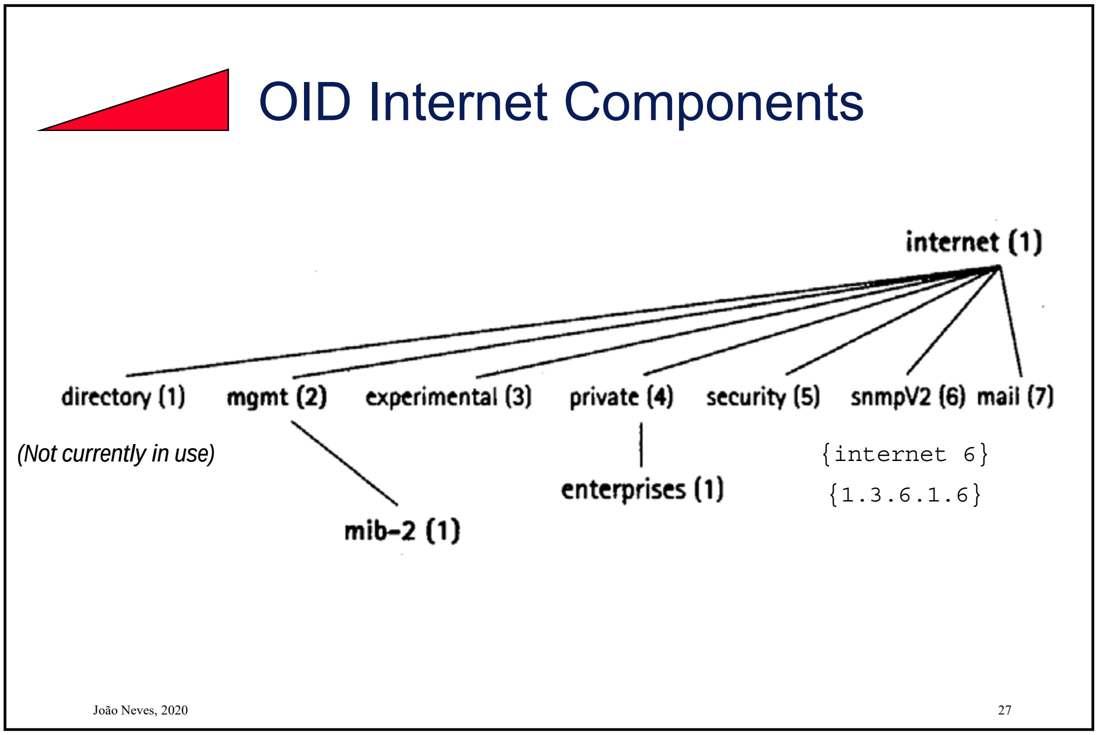
```
internet      OBJECT IDENTIFIER ::= { iso 3 6 1 }
```

That is, the subtree of Internet OBJECT IDENTIFIER starts with:

```
              1.3.6.1
```

Several branches of this part of the tree are used for network management:

```
mgmt          OBJECT IDENTIFIER ::= { internet 2 }
experimental  OBJECT IDENTIFIER ::= { internet 3 }
private       OBJECT IDENTIFIER ::= { internet 4 }
enterprises   OBJECT IDENTIFIER ::= { private 1 }
```

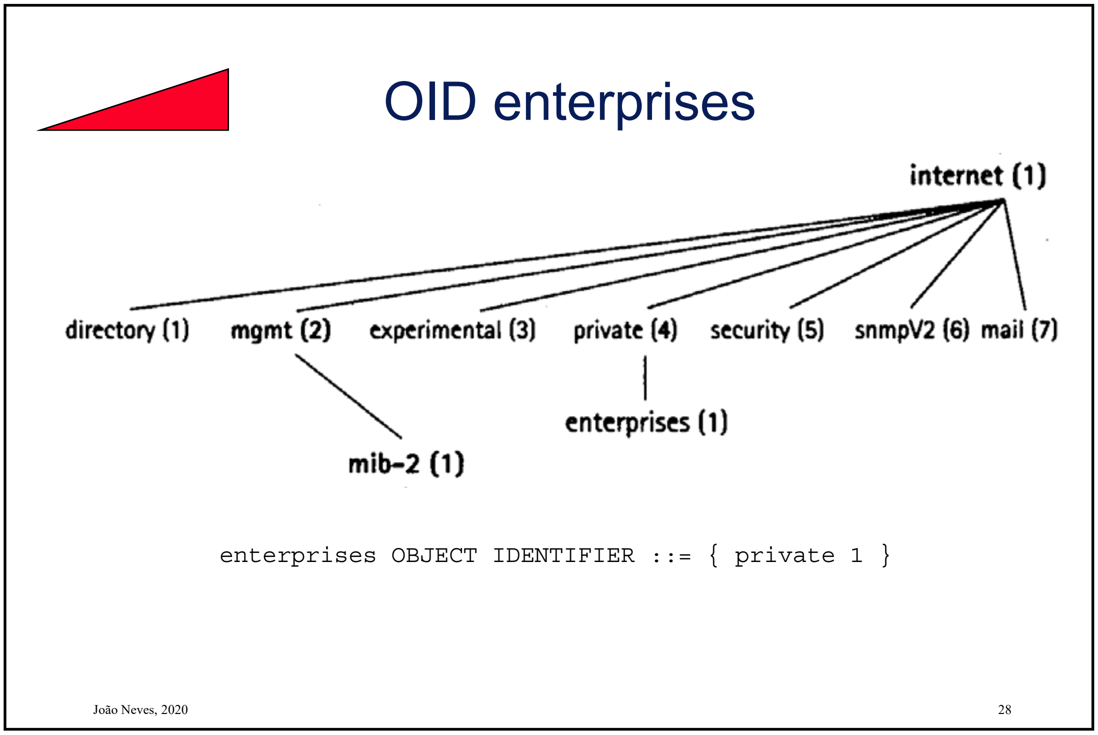


## OID Internet Components

*(Not currently in use)*

{internet 6}

{1.3.6.1.6}



## OID enterprises

```
enterprises OBJECT IDENTIFIER ::= { private 1 }
```

# OID enterprises

- The list of all numbers assigned to the **`enterprises`** variable is managed by the “Internet Assigned Numbers Authority” (IANA) and can be obtained from:

`http://www.iana.org/assignments/enterprise-numbers`

- Under their private subtree, manufacturers can put the structure they see fit

- For example, the Cisco System has `enterprises` OID 9, i.e.

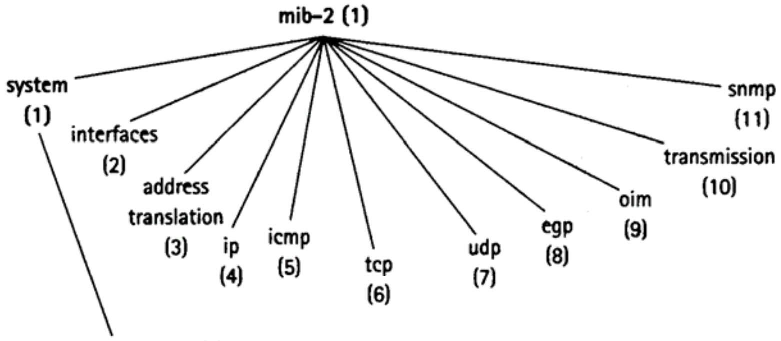
`{ iso.org.dod.internet.private.enterprises.cisco }`

or `{ 1.3.6.1.4.1.9 }`

## http://www.iana.org/assignments/enterprise-numbers

```
PRIVATE ENTERPRISE NUMBERS

SMI Network Management Private Enterprise Codes:

Prefix: iso.org.dod.internet.private.enterprise (1.3.6.1.4.1)

Decimal   Name                                                References
-------   ----                                                ----------
      0   Reserved                      Joyce K. Reynolds  jkrey@isi.edu
      1   NxNetworks            Michael Kellen  OID.Admin@NxNetworks.com
      2   IBM               Bob Moore                 remoore@us.ibm.com
      3   CMU                     Steve Waldbusser  sw01+@andrew.cmu.edu
      4   Unix              Keith Sklower   sklower@okeeffe.berkeley.edu
      5   ACC                           Art Berggreen   art@SALT.ACC.COM
      6   TWG          John Lunny  jlunny@eco.twg.com   (603) 847-4500
      7   CAYMAN                          Beth Miaoulis  beth@cayman.com
      8   PSI                   Marty Schoffstahl  schoff@NISC.NYSER.NET
      9   cisco                                Greg Satz  satz@CISCO.COM
     10   NSC                       John Lyman         lyman@network.com
     11   Hewlett Packard  R. Dwight Schettler  rds%hpcndm@HPLABS.HP.COM
     12   Epilogue                     Karl Auerbach   karl@cavebear.com
     13   U of Tennessee                  Jeffrey Case   case@CS.UTK.EDU
     14   BBN          Arif Diwan     adiwan@bbn.com, bbn-mibs@bbn.com
     15   Xylogics, Inc.          Jim Barnes         barnes@xylogics.com
     16   Timeplex                      Laura Bridge  laura@uunet.UU.NET
     17   Canstar                   Sanand Patel  sanand@HUB.TORONTO.EDU
     18   Wellfleet                   Caralyn Brown cbrown@wellfleet.com

     [...]
```

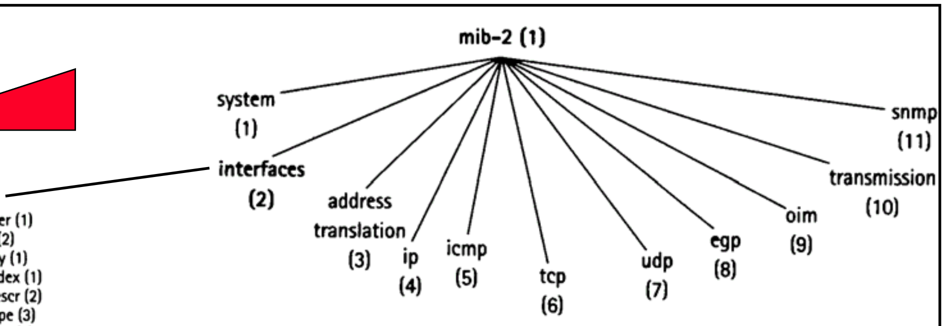
# The Group *system*



sysDescr (1)
sysObjectID (2)
sysUpTime (3)
sysContact* (4)
sysName* (5)
sysLocation* (6)
sysServices* (7)

* Adicionado na MIB-II

João Neves, 2020 31

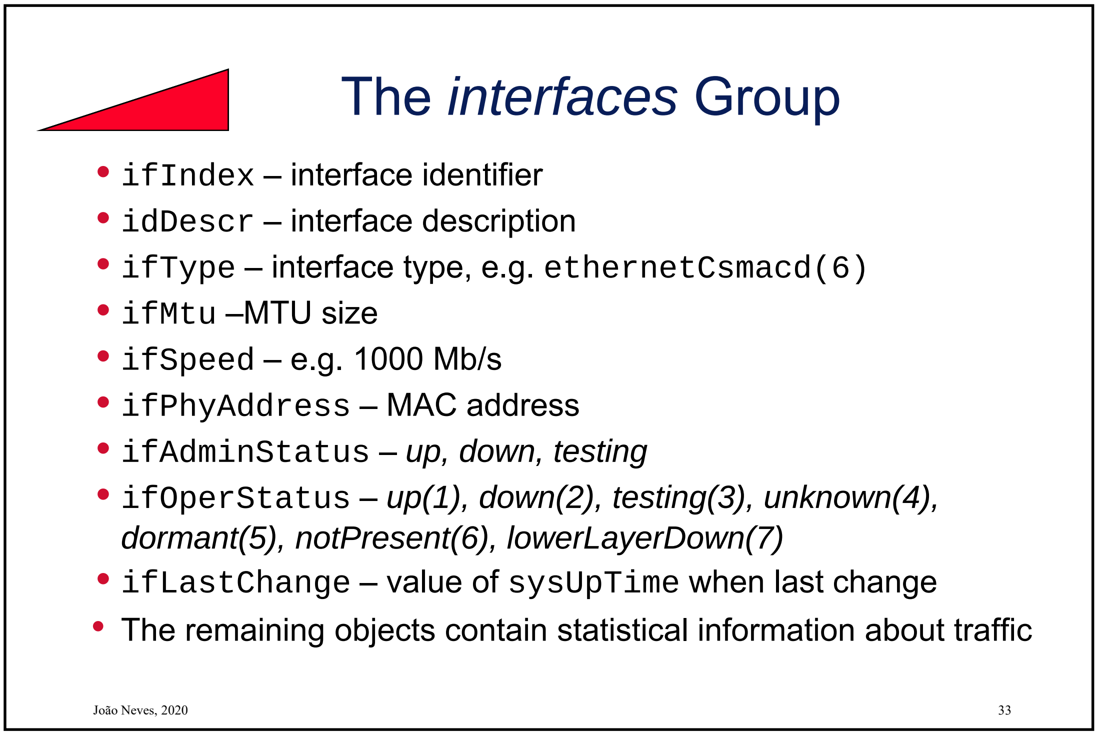


ifNumber (1)
ifTable (2)
ifEntry (1)
ifIndex (1)
ifDescr (2)
ifType (3)
ifMtu (4)
ifSpeed (5)
ifPhysicalAddress (6)
ifAdminStatus (7)
ifOperStatus (8)
ifLastChange (9)
ifInOctets (10)
ifInUcastPkts (11)
ifInNUcastPkts (12)
ifInDiscards (13)
ifInErrors (14)
ifInUnknownProtos (15)
ifOutOctets (16)
ifOutUcastPkts (17)
ifOutNUcastPkts (18)
ifOutDiscards (19)
ifOutErrors (20)
ifOutQLen (21)
ifSpecific* (22)

# The Group *interfaces*

- The *interfaces* group provides information about the hardware interfaces of a managed system; this information is organized in a table;
- The first object (`ifNumber`) indicates the number of system interfaces;
- For each interface a row is created in the table, with 22 columns.

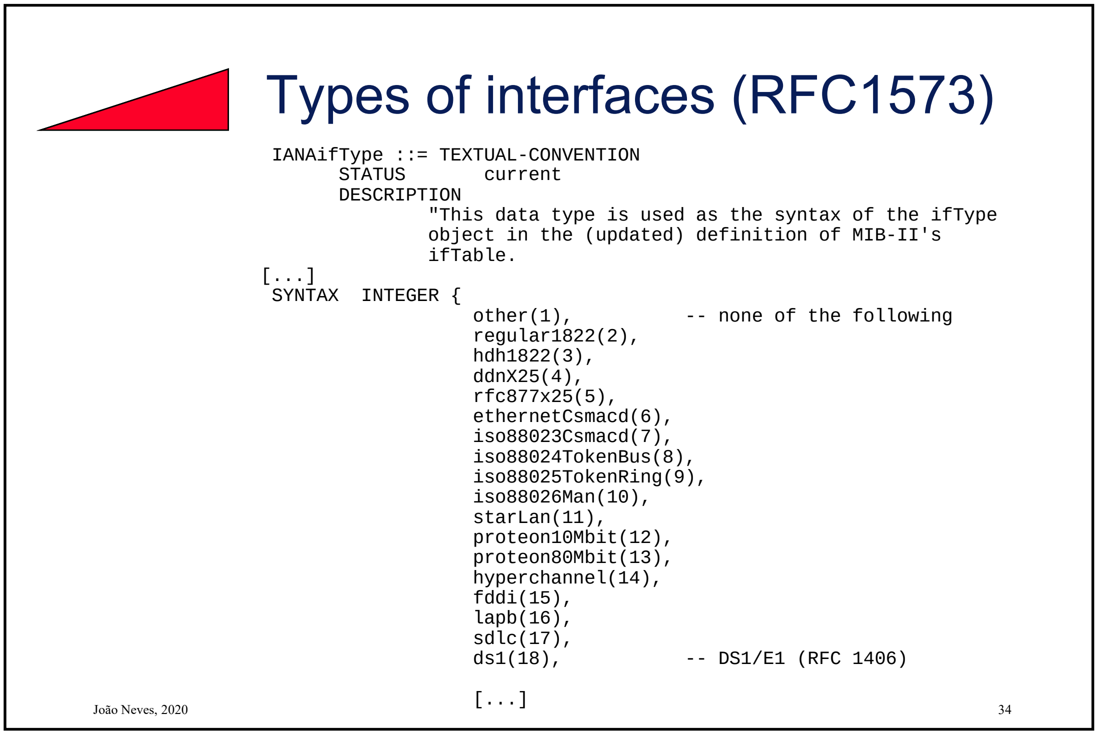
João Neves, 2020 32



## The *interfaces* Group

- `ifIndex` – interface identifier
- `idDescr` – interface description
- `ifType` – interface type, e.g. `ethernetCsmacd(6)`
- `ifMtu` –MTU size
- `ifSpeed` – e.g. 1000 Mb/s
- `ifPhyAddress` – MAC address
- `ifAdminStatus` – *up, down, testing*
- `ifOperStatus` – *up(1), down(2), testing(3), unknown(4), dormant(5), notPresent(6), lowerLayerDown(7)*
- `ifLastChange` – value of `sysUpTime` when last change
- The remaining objects contain statistical information about traffic

João Neves, 2020 33



## Types of interfaces (RFC1573)

```
 IANAifType ::= TEXTUAL-CONVENTION
       STATUS       current
       DESCRIPTION
               "This data type is used as the syntax of the ifType
               object in the (updated) definition of MIB-II's
               ifTable.
[...]
 SYNTAX  INTEGER {
                   other(1),          -- none of the following
                   regular1822(2),
                   hdh1822(3),
                   ddnX25(4),
                   rfc877x25(5),
                   ethernetCsmacd(6),
                   iso88023Csmacd(7),
                   iso88024TokenBus(8),
                   iso88025TokenRing(9),
                   iso88026Man(10),
                   starLan(11),
                   proteon10Mbit(12),
                   proteon80Mbit(13),
                   hyperchannel(14),
                   fddi(15),
                   lapb(16),
                   sdlc(17),
                   ds1(18),           -- DS1/E1 (RFC 1406)

                   [...]
```

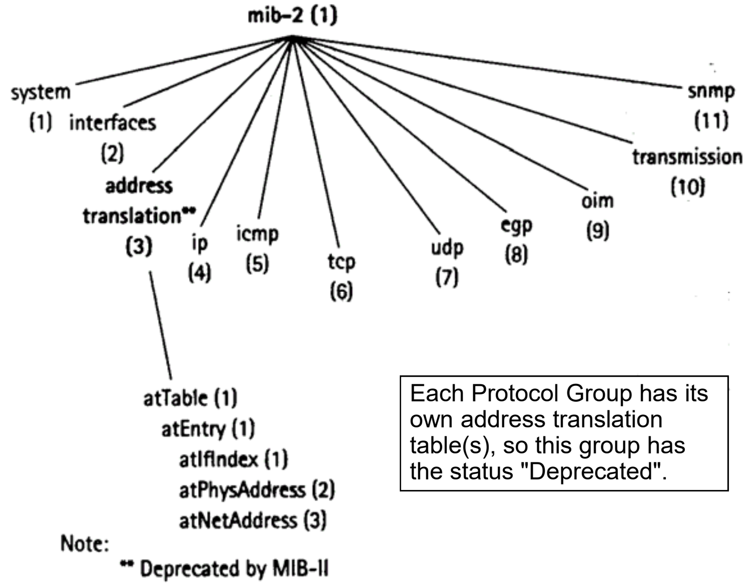
João Neves, 2020 34

## Types of interfaces (cont.)

```
[...]
                  e1(19),            -- obsolete
                  basicISDN(20),
                  primaryISDN(21),
                  propPointToPointSerial(22), -- proprietary serial
                  ppp(23),
                  softwareLoopback(24),
                  eon(25),            -- CLNP over IP (RFC 1070)
                  ethernet3Mbit(26),
                  nsip(27),           -- XNS over IP
                  slip(28),           -- generic SLIP
                  ultra(29),          -- ULTRA technologies
                  ds3(30),            -- T-3
                  sip(31),            -- SMDS
                  frameRelay(32),    -- DTE only
                  rs232(33),
                  para(34),           -- parallel-port
                  arcnet(35),         -- arcnet
                  arcnetPlus(36),     -- arcnet plus
                  atm(37),            -- ATM cells
                  miox25(38),
                  sonet(39),          -- SONET or SDH
                  x25ple(40),
                  iso88022llc(41),
                  localTalk(42),
                  smdsDxi(43),
                  frameRelayService(44),  -- Frame relay DCE
                  v35(45),
                  hssi(46),
                  hippi(47),
                  modem(48),          -- Generic modem
                  aal5(49),           -- AAL5 over ATM
                  sonetPath(50),
                  sonetVT(51),
                  smdsIcip(52),       -- SMDS InterCarrier Interface
                  propVirtual(53),    -- proprietary virtual/internal
                  propMultiplexor(54) -- proprietary multiplexing
              }

   END
```

João Neves, 2020 35

## The *address translation* Group



Each Protocol Group has its own address translation table(s), so this group has the status "Deprecated".

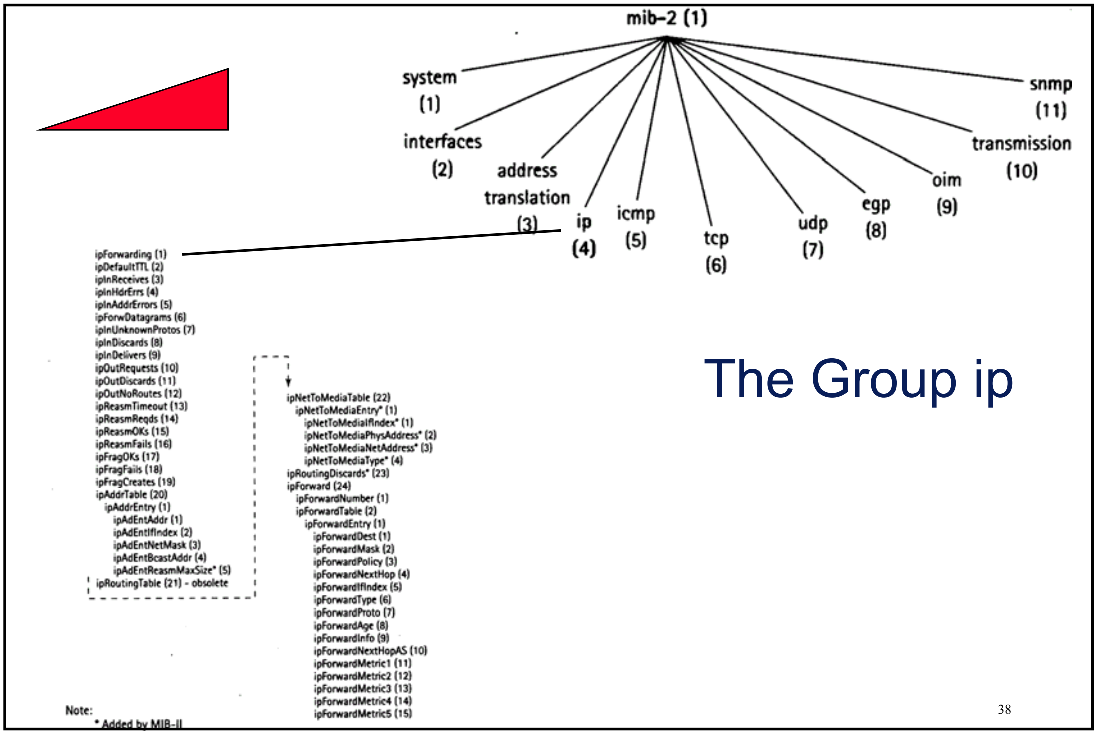
João Neves, 2020

# The atTable Table

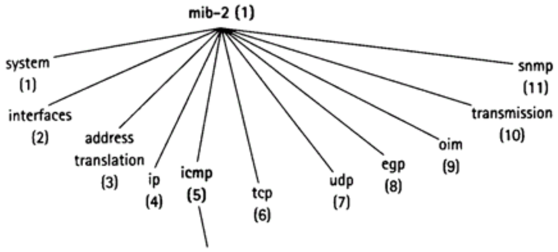
| atIfIndex | atPhysAddress | atNetAddress |
|---|---|---|
| 1 | | aa.bb.cc.dd |
| 2 | | ee.ff.gg.hh |
| | | |
| n | | ww.xx.yy.zz |

- `atIfIndex` – identifies the interface where the addresses are valid
- `atPhysAddress` – may be a MAC address
- `atNetAddress` – for example an IP address

# The Group ip



# icmp

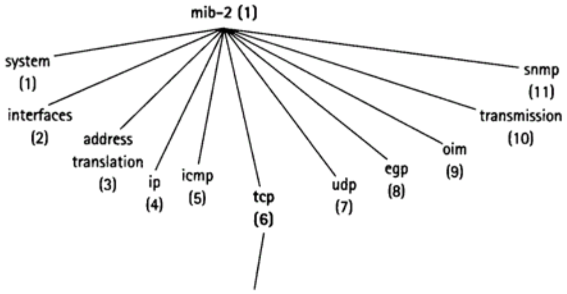


icmpInMsgs (1)
icmpInErrors (2)
icmpInDestUnreachs (3)
icmpInTimeExcds (4)
icmpInParmProbs (5)
icmpInSrcQuenchs (6)
icmpInRedirects (7)
icmpInEchos (8)
IcmpInEchoReps (9)
icmpInTimeStamps (10)
icmpInTimestampReps (11)
icmpInAddrMasks (12)
icmpInAddrMaskReps (13)
icmpOutMsgs (14)
icmpOutErrors (15)
icmpOutDestUnreachs (16)
icmpOutTimeExcds (17)
icmpOutParmProbs (18)
icmpOutSrcQuenchs (19)
icmpOutRedirects (20)
icmpOutEchos (21)
icmpOutEchoReps (22)
icmpOutTimeStamps (23)
icmpOutTimestampReps (24)
icmpOutAddrMasks (25)
icmpOutAddrMaskReps (26)

João Neves, 2020

39

# The Group tcp



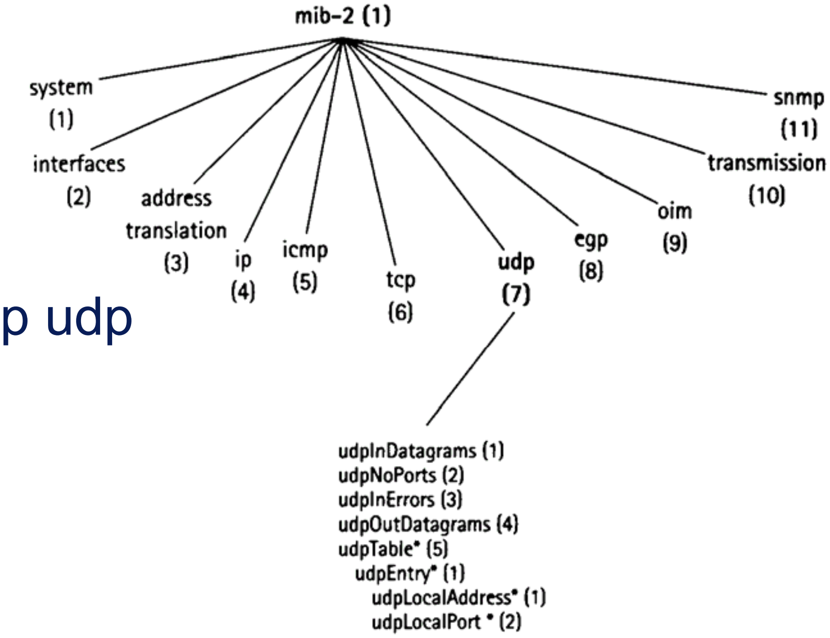
tcpRtoAlgorithm (1)
tcpRtoMin (2)
tcpRtoMax (3)
tcpMaxConn (4)
tcpActiveOpens (5)
tcpPassiveOpens (6)
tcpAttemptFails (7)
tcpEstabResets (8)
tcpCurrEstab (9)
tcpInSegs (10)
tcpOutSegs (11)
tcpRetransSegs (12)
tcpConnTable (13)
  tcpConnEntry (1)
    tcpConnState (1)
    tcpConnLocalAddress (2)
    tcpConnLocalPort (3)
    tcpConnRemAddress (4)
    tcpConnRemPort (5)
tcpInErrs* (14)
tcpOutRsts* (15)

Note:
* Added by MIB-II

João Neves, 2020

40

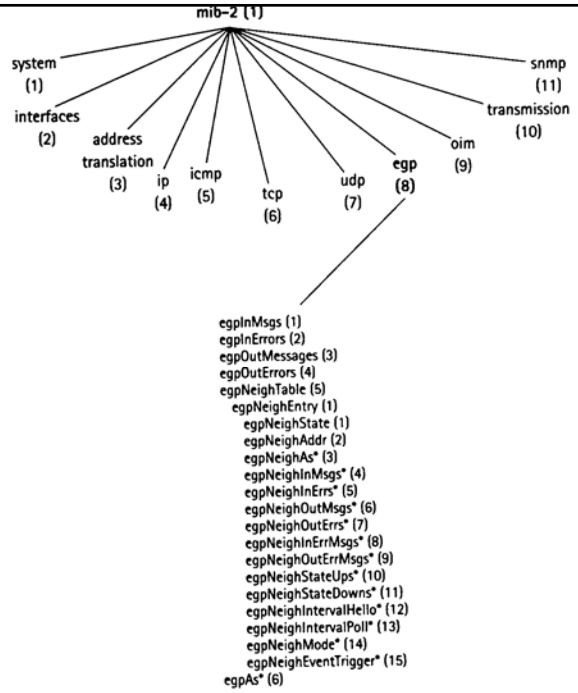
## The Group udp



udpInDatagrams (1)
udpNoPorts (2)
udpInErrors (3)
udpOutDatagrams (4)
udpTable* (5)
  udpEntry* (1)
    udpLocalAddress* (1)
    udpLocalPort * (2)

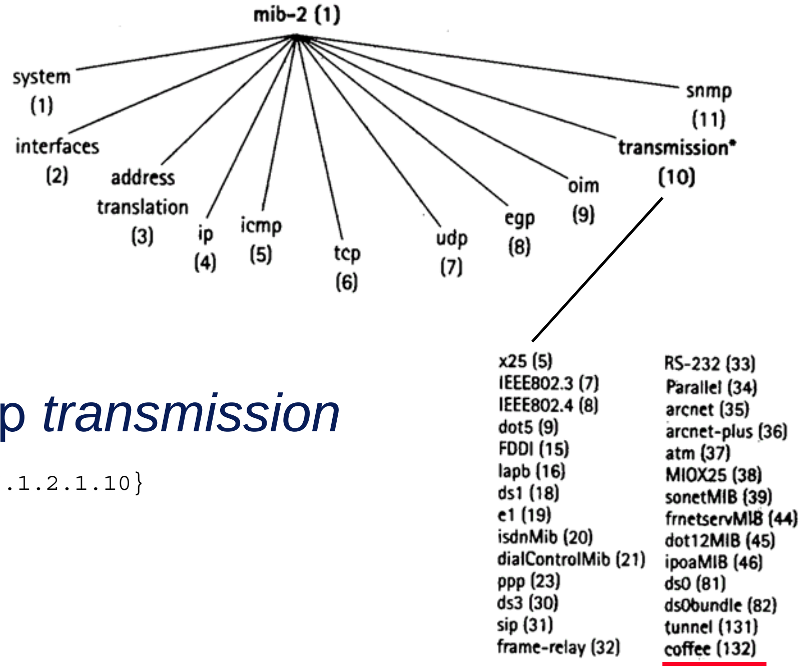
Note:
* Added by MIB-II

## The Group egp



egpInMsgs (1)
egpInErrors (2)
egpOutMessages (3)
egpOutErrors (4)
egpNeighTable (5)
  egpNeighEntry (1)
    egpNeighState (1)
    egpNeighAddr (2)
    egpNeighAs* (3)
    egpNeighInMsgs* (4)
    egpNeighInErrs* (5)
    egpNeighOutMsgs* (6)
    egpNeighOutErrs* (7)
    egpNeighInErrMsgs* (8)
    egpNeighOutErrMsgs* (9)
    egpNeighStateUps* (10)
    egpNeighStateDowns* (11)
    egpNeighIntervalHello* (12)
    egpNeighIntervalPoll* (13)
    egpNeighMode* (14)
    egpNeighEventTrigger* (15)
egpAs* (6)

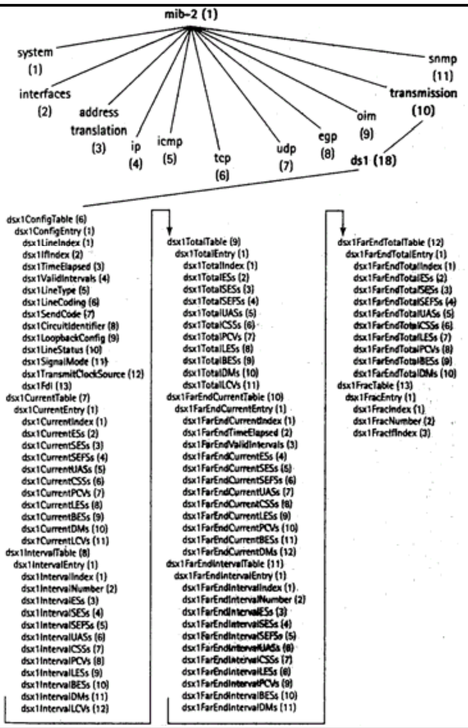
Note:
* Added by MIB-II



# The Group *transmission*

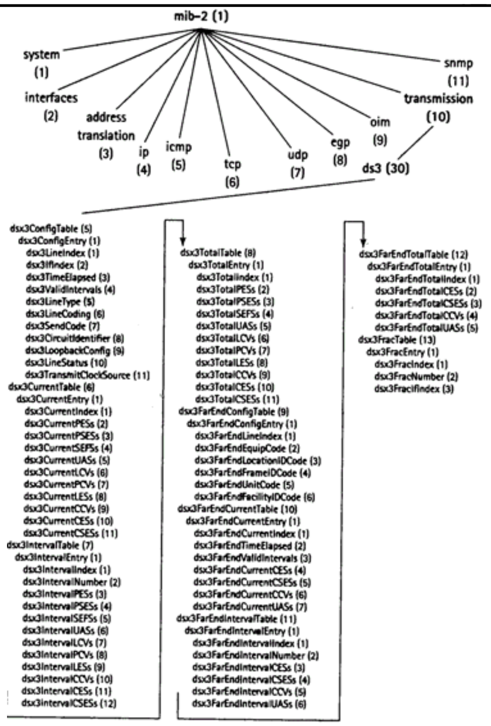
{1.3.6.1.2.1.10}

* Added by MIB-II



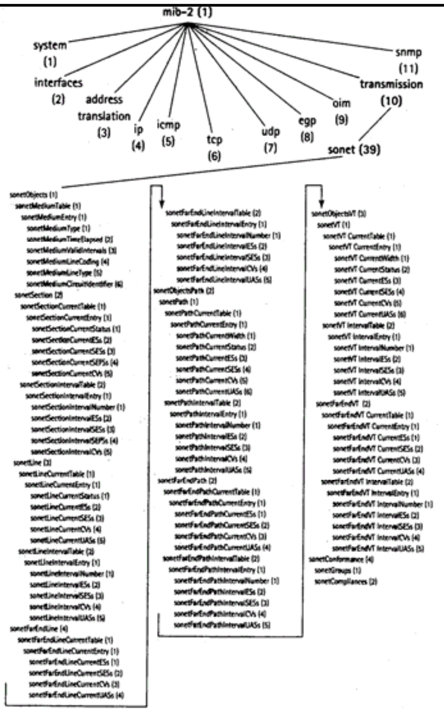
# The DS1/E1 Objects

## The DS3/E3 Objects



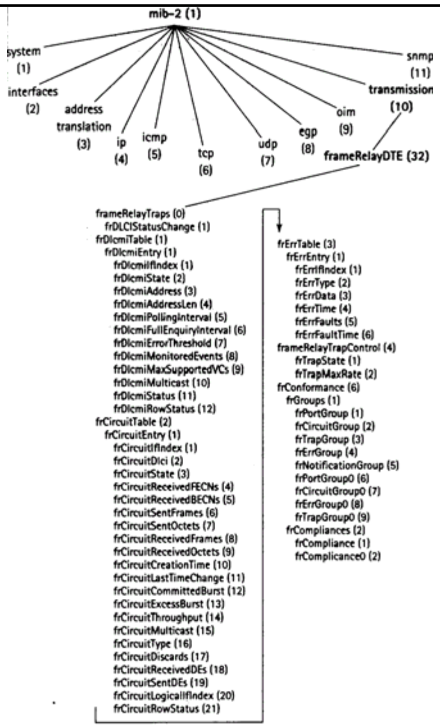
```
mib-2 (1)
system (1)
interfaces (2)
address translation (3)
ip (4)
icmp (5)
tcp (6)
udp (7)
egp (8)
oim (9)
transmission (10)
snmp (11)
ds3 (30)

dsx3ConfigTable (5)
  dsx3ConfigEntry (1)
    dsx3LineIndex (1)
    dsx3IfIndex (2)
    dsx3TimeElapsed (3)
    dsx3ValidIntervals (4)
    dsx3LineType (5)
    dsx3LineCoding (6)
    dsx3SendCode (7)
    dsx3CircuitIdentifier (8)
    dsx3LoopbackConfig (9)
    dsx3LineStatus (10)
    dsx3TransmitClockSource (11)
dsx3CurrentTable (6)
  dsx3CurrentEntry (1)
    dsx3CurrentIndex (1)
    dsx3CurrentPESs (2)
    dsx3CurrentPSESs (3)
    dsx3CurrentSEFSs (4)
    dsx3CurrentUASs (5)
    dsx3CurrentLCVs (6)
    dsx3CurrentPCVs (7)
    dsx3CurrentLESs (8)
    dsx3CurrentCCVs (9)
    dsx3CurrentCESs (10)
    dsx3CurrentCSESs (11)
dsx3IntervalTable (7)
  dsx3IntervalEntry (1)
    dsx3IntervalIndex (1)
    dsx3IntervalNumber (2)
    dsx3IntervalPESs (3)
    dsx3IntervalPSESs (4)
    dsx3IntervalSEFSs (5)
    dsx3IntervalUASs (6)
    dsx3IntervalLCVs (7)
    dsx3IntervalPCVs (8)
    dsx3IntervalLESs (9)
    dsx3IntervalCCVs (10)
    dsx3IntervalCESs (11)
    dsx3IntervalCSESs (12)

dsx3TotalTable (8)
  dsx3TotalEntry (1)
    dsx3TotalIndex (1)
    dsx3TotalPESs (2)
    dsx3TotalPSESs (3)
    dsx3TotalSEFSs (4)
    dsx3TotalUASs (5)
    dsx3TotalLCVs (6)
    dsx3TotalPCVs (7)
    dsx3TotalLESs (8)
    dsx3TotalCCVs (9)
    dsx3TotalCESs (10)
    dsx3TotalCSESs (11)
dsx3FarEndConfigTable (9)
  dsx3FarEndConfigEntry (1)
    dsx3FarEndLineIndex (1)
    dsx3FarEndEquipCode (2)
    dsx3FarEndLocationIDCode (3)
    dsx3FarEndFrameIDCode (4)
    dsx3FarEndUnitCode (5)
    dsx3FarEndFacilityIDCode (6)
dsx3FarEndCurrentTable (10)
  dsx3FarEndCurrentEntry (1)
    dsx3FarEndCurrentIndex (1)
    dsx3FarEndTimeElapsed (2)
    dsx3FarEndValidIntervals (3)
    dsx3FarEndCurrentCESs (4)
    dsx3FarEndCurrentCSESs (5)
    dsx3FarEndCurrentCCVs (6)
    dsx3FarEndCurrentUASs (7)
dsx3FarEndIntervalTable (11)
  dsx3FarEndIntervalEntry (1)
    dsx3FarEndIntervalIndex (1)
    dsx3FarEndIntervalNumber (2)
    dsx3FarEndIntervalCESs (3)
    dsx3FarEndIntervalCSESs (4)
    dsx3FarEndIntervalCCVs (5)
    dsx3FarEndIntervalUASs (6)

dsx3FarEndTotalTable (12)
  dsx3FarEndTotalEntry (1)
    dsx3FarEndTotalIndex (1)
    dsx3FarEndTotalCESs (2)
    dsx3FarEndTotalCSESs (3)
    dsx3FarEndTotalCCVs (4)
    dsx3FarEndTotalUASs (5)
dsx3FracTable (13)
  dsx3FracEntry (1)
    dsx3FracIndex (1)
    dsx3FracNumber (2)
    dsx3FracIfIndex (3)
```

## The SONET Objects



```
mib-2 (1)
system (1)
interfaces (2)
address translation (3)
ip (4)
icmp (5)
tcp (6)
udp (7)
egp (8)
oim (9)
transmission (10)
snmp (11)
sonet (39)

sonetObjects (1)
  sonetMediumTable (1)
    sonetMediumEntry (1)
      sonetMediumType (1)
      sonetMediumTimeElapsed (2)
      sonetMediumValidIntervals (3)
      sonetMediumLineCoding (4)
      sonetMediumLineType (5)
      sonetMediumCircuitIdentifier (6)
  sonetSection (2)
    sonetSectionCurrentTable (1)
      sonetSectionCurrentEntry (1)
        sonetSectionCurrentStatus (1)
        sonetSectionCurrentESs (2)
        sonetSectionCurrentSESs (3)
        sonetSectionCurrentSEFSs (4)
        sonetSectionCurrentCVs (5)
    sonetSectionIntervalTable (2)
      sonetSectionIntervalEntry (1)
        sonetSectionIntervalNumber (1)
        sonetSectionIntervalESs (2)
        sonetSectionIntervalSESs (3)
        sonetSectionIntervalSEFSs (4)
        sonetSectionIntervalCVs (5)
  sonetLine (3)
    sonetLineCurrentTable (1)
      sonetLineCurrentEntry (1)
        sonetLineCurrentStatus (1)
        sonetLineCurrentESs (2)
        sonetLineCurrentSESs (3)
        sonetLineCurrentCVs (4)
        sonetLineCurrentUASs (5)
    sonetLineIntervalTable (2)
      sonetLineIntervalEntry (1)
        sonetLineIntervalNumber (1)
        sonetLineIntervalESs (2)
        sonetLineIntervalSESs (3)
        sonetLineIntervalCVs (4)
        sonetLineIntervalUASs (5)
  sonetFarEndLine (4)
    sonetFarEndLineCurrentTable (1)
      sonetFarEndLineCurrentEntry (1)
        sonetFarEndLineCurrentESs (1)
        sonetFarEndLineCurrentSESs (2)
        sonetFarEndLineCurrentCVs (3)
        sonetFarEndLineCurrentUASs (4)

    sonetFarEndLineIntervalTable (2)
      sonetFarEndLineIntervalEntry (1)
        sonetFarEndLineIntervalNumber (1)
        sonetFarEndLineIntervalESs (2)
        sonetFarEndLineIntervalSESs (3)
        sonetFarEndLineIntervalCVs (4)
        sonetFarEndLineIntervalUASs (5)
sonetObjectsPath (2)
  sonetPath (1)
    sonetPathCurrentTable (1)
      sonetPathCurrentEntry (1)
        sonetPathCurrentWidth (1)
        sonetPathCurrentStatus (2)
        sonetPathCurrentESs (3)
        sonetPathCurrentSESs (4)
        sonetPathCurrentCVs (5)
        sonetPathCurrentUASs (6)
    sonetPathIntervalTable (2)
      sonetPathIntervalEntry (1)
        sonetPathIntervalNumber (1)
        sonetPathIntervalESs (2)
        sonetPathIntervalSESs (3)
        sonetPathIntervalCVs (4)
        sonetPathIntervalUASs (5)
  sonetFarEndPath (2)
    sonetFarEndPathCurrentTable (1)
      sonetFarEndPathCurrentEntry (1)
        sonetFarEndPathCurrentESs (1)
        sonetFarEndPathCurrentSESs (2)
        sonetFarEndPathCurrentCVs (3)
        sonetFarEndPathCurrentUASs (4)
    sonetFarEndPathIntervalTable (2)
      sonetFarEndPathIntervalEntry (1)
        sonetFarEndPathIntervalNumber (1)
        sonetFarEndPathIntervalESs (2)
        sonetFarEndPathIntervalSESs (3)
        sonetFarEndPathIntervalCVs (4)
        sonetFarEndPathIntervalUASs (5)

sonetObjectsVT (3)
  sonetVT (1)
    sonetVT CurrentTable (1)
      sonetVT CurrentEntry (1)
        sonetVT CurrentWidth (1)
        sonetVT CurrentStatus (2)
        sonetVT CurrentESs (3)
        sonetVT CurrentSESs (4)
        sonetVT CurrentCVs (5)
        sonetVT CurrentUASs (6)
    sonetVT IntervalTable (2)
      sonetVT IntervalEntry (1)
        sonetVT IntervalNumber (1)
        sonetVT IntervalESs (2)
        sonetVT IntervalSESs (3)
        sonetVT IntervalCVs (4)
        sonetVT IntervalUASs (5)
  sonetFarEndVT (2)
    sonetFarEndVT CurrentTable (1)
      sonetFarEndVT CurrentEntry (1)
        sonetFarEndVT CurrentESs (1)
        sonetFarEndVT CurrentSESs (2)
        sonetFarEndVT CurrentCVs (3)
        sonetFarEndVT CurrentUASs (4)
    sonetFarEndVT IntervalTable (2)
      sonetFarEndVT IntervalEntry (1)
        sonetFarEndVT IntervalNumber (1)
        sonetFarEndVT IntervalESs (2)
        sonetFarEndVT IntervalSESs (3)
        sonetFarEndVT IntervalCVs (4)
        sonetFarEndVT IntervalUASs (5)
sonetConformance (4)
  sonetGroups (1)
  sonetCompliances (2)
```

## The Frame Relay DTE Objects



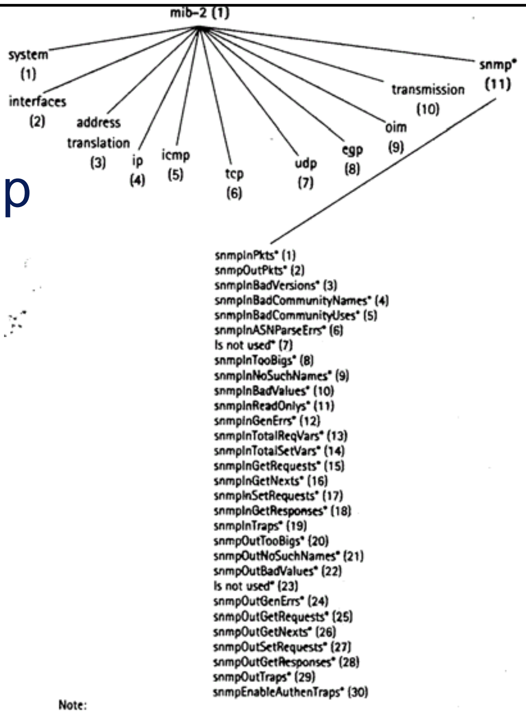
mib-2 (1)

system (1)
interfaces (2)
address translation (3)
ip (4)
icmp (5)
tcp (6)
udp (7)
egp (8)
oim (9)
transmission (10)
snmp (11)

frameRelayDTE (32)

- frameRelayTraps (0)
  - frDLCIStatusChange (1)
- frDlcmiTable (1)
  - frDlcmiEntry (1)
    - frDlcmiIfIndex (1)
    - frDlcmiState (2)
    - frDlcmiAddress (3)
    - frDlcmiAddressLen (4)
    - frDlcmiPollingInterval (5)
    - frDlcmiFullEnquiryInterval (6)
    - frDlcmiErrorThreshold (7)
    - frDlcmiMonitoredEvents (8)
    - frDlcmiMaxSupportedVCs (9)
    - frDlcmiMulticast (10)
    - frDlcmiStatus (11)
    - frDlcmiRowStatus (12)
- frCircuitTable (2)
  - frCircuitEntry (1)
    - frCircuitIfIndex (1)
    - frCircuitDlci (2)
    - frCircuitState (3)
    - frCircuitReceivedFECNs (4)
    - frCircuitReceivedBECNs (5)
    - frCircuitSentFrames (6)
    - frCircuitSentOctets (7)
    - frCircuitReceivedFrames (8)
    - frCircuitReceivedOctets (9)
    - frCircuitCreationTime (10)
    - frCircuitLastTimeChange (11)
    - frCircuitCommittedBurst (12)
    - frCircuitExcessBurst (13)
    - frCircuitThroughput (14)
    - frCircuitMulticast (15)
    - frCircuitType (16)
    - frCircuitDiscards (17)
    - frCircuitReceivedDEs (18)
    - frCircuitSentDEs (19)
    - frCircuitLogicalIfIndex (20)
    - frCircuitRowStatus (21)
- frErrTable (3)
  - frErrEntry (1)
    - frErrIfIndex (1)
    - frErrType (2)
    - frErrData (3)
    - frErrTime (4)
    - frErrFaults (5)
    - frErrFaultTime (6)
- frameRelayTrapControl (4)
  - frTrapState (1)
  - frTrapMaxRate (2)
- frConformance (6)
  - frGroups (1)
    - frPortGroup (1)
    - frCircuitGroup (2)
    - frTrapGroup (3)
    - frErrGroup (4)
    - frNotificationGroup (5)
    - frPortGroup0 (6)
    - frCircuitGroup0 (7)
    - frErrGroup0 (8)
    - frTrapGroup0 (9)
  - frCompliances (2)
    - frCompliance (1)
    - frComplicance0 (2)

## The SNMP Group

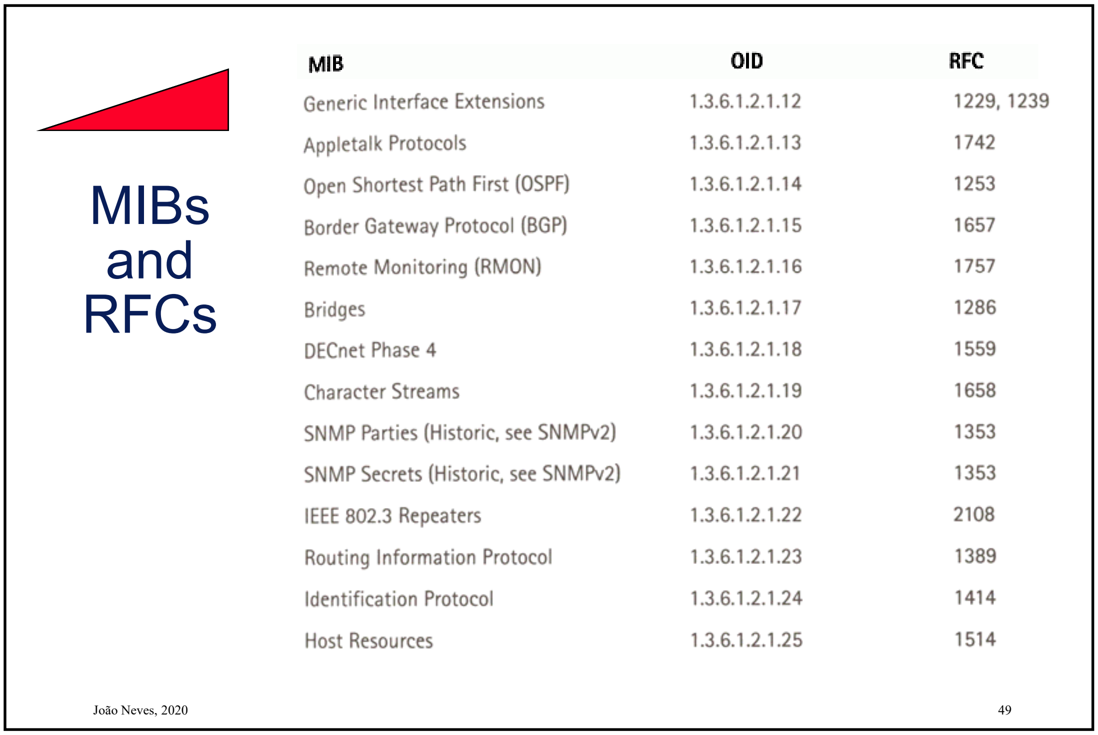


mib-2 (1)

system (1)
interfaces (2)
address translation (3)
ip (4)
icmp (5)
tcp (6)
udp (7)
egp (8)
oim (9)
transmission (10)
snmp* (11)

snmpInPkts* (1)
snmpOutPkts* (2)
snmpInBadVersions* (3)
snmpInBadCommunityNames* (4)
snmpInBadCommunityUses* (5)
snmpInASNParseErrs* (6)
Is not used* (7)
snmpInTooBigs* (8)
snmpInNoSuchNames* (9)
snmpInBadValues* (10)
snmpInReadOnlys* (11)
snmpInGenErrs* (12)
snmpInTotalReqVars* (13)
snmpInTotalSetVars* (14)
snmpInGetRequests* (15)
snmpInGetNexts* (16)
snmpInSetRequests* (17)
snmpInGetResponses* (18)
snmpInTraps* (19)
snmpOutTooBigs* (20)
snmpOutNoSuchNames* (21)
snmpOutBadValues* (22)
Is not used* (23)
snmpOutGenErrs* (24)
snmpOutGetRequests* (25)
snmpOutGetNexts* (26)
snmpOutSetRequests* (27)
snmpOutGetResponses* (28)
snmpOutTraps* (29)
snmpEnableAuthenTraps* (30)

Note:
* Added by MIB-II

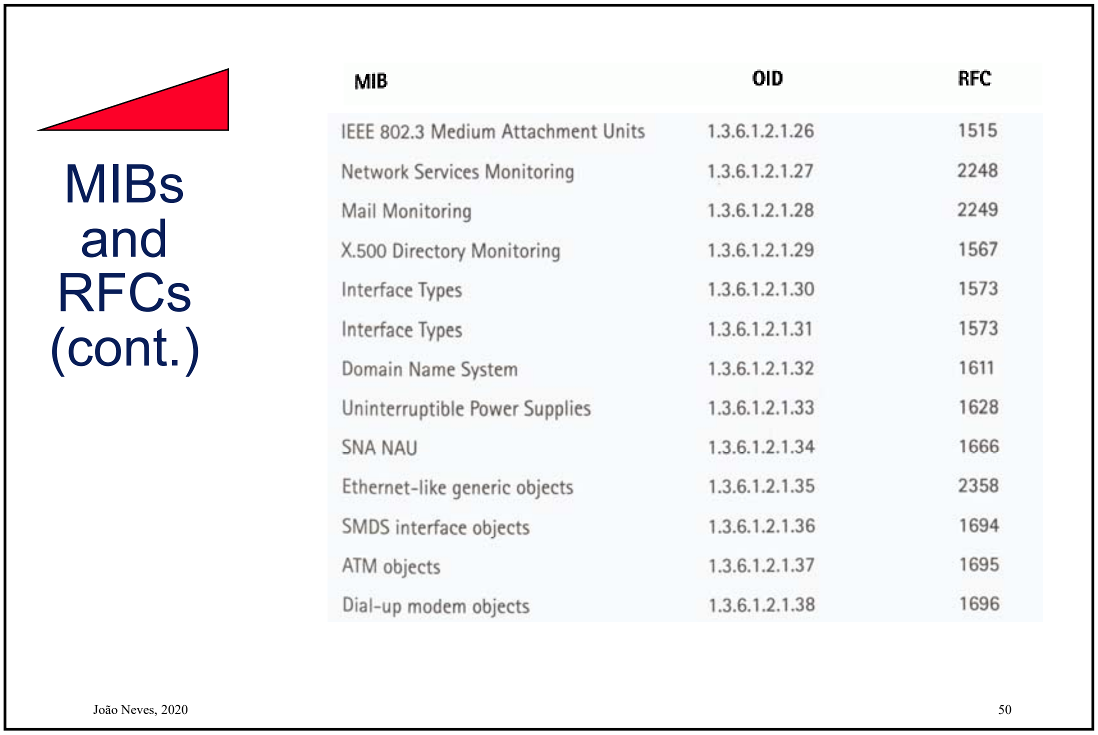


## MIBs and RFCs

| MIB | OID | RFC |
|---|---|---|
| Generic Interface Extensions | 1.3.6.1.2.1.12 | 1229, 1239 |
| Appletalk Protocols | 1.3.6.1.2.1.13 | 1742 |
| Open Shortest Path First (OSPF) | 1.3.6.1.2.1.14 | 1253 |
| Border Gateway Protocol (BGP) | 1.3.6.1.2.1.15 | 1657 |
| Remote Monitoring (RMON) | 1.3.6.1.2.1.16 | 1757 |
| Bridges | 1.3.6.1.2.1.17 | 1286 |
| DECnet Phase 4 | 1.3.6.1.2.1.18 | 1559 |
| Character Streams | 1.3.6.1.2.1.19 | 1658 |
| SNMP Parties (Historic, see SNMPv2) | 1.3.6.1.2.1.20 | 1353 |
| SNMP Secrets (Historic, see SNMPv2) | 1.3.6.1.2.1.21 | 1353 |
| IEEE 802.3 Repeaters | 1.3.6.1.2.1.22 | 2108 |
| Routing Information Protocol | 1.3.6.1.2.1.23 | 1389 |
| Identification Protocol | 1.3.6.1.2.1.24 | 1414 |
| Host Resources | 1.3.6.1.2.1.25 | 1514 |

João Neves, 2020

49



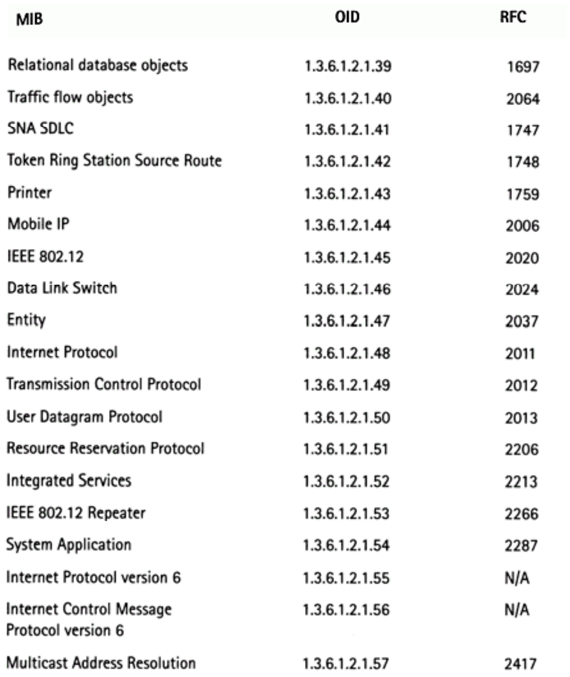
## MIBs and RFCs (cont.)

| MIB | OID | RFC |
|---|---|---|
| IEEE 802.3 Medium Attachment Units | 1.3.6.1.2.1.26 | 1515 |
| Network Services Monitoring | 1.3.6.1.2.1.27 | 2248 |
| Mail Monitoring | 1.3.6.1.2.1.28 | 2249 |
| X.500 Directory Monitoring | 1.3.6.1.2.1.29 | 1567 |
| Interface Types | 1.3.6.1.2.1.30 | 1573 |
| Interface Types | 1.3.6.1.2.1.31 | 1573 |
| Domain Name System | 1.3.6.1.2.1.32 | 1611 |
| Uninterruptible Power Supplies | 1.3.6.1.2.1.33 | 1628 |
| SNA NAU | 1.3.6.1.2.1.34 | 1666 |
| Ethernet-like generic objects | 1.3.6.1.2.1.35 | 2358 |
| SMDS interface objects | 1.3.6.1.2.1.36 | 1694 |
| ATM objects | 1.3.6.1.2.1.37 | 1695 |
| Dial-up modem objects | 1.3.6.1.2.1.38 | 1696 |

João Neves, 2020

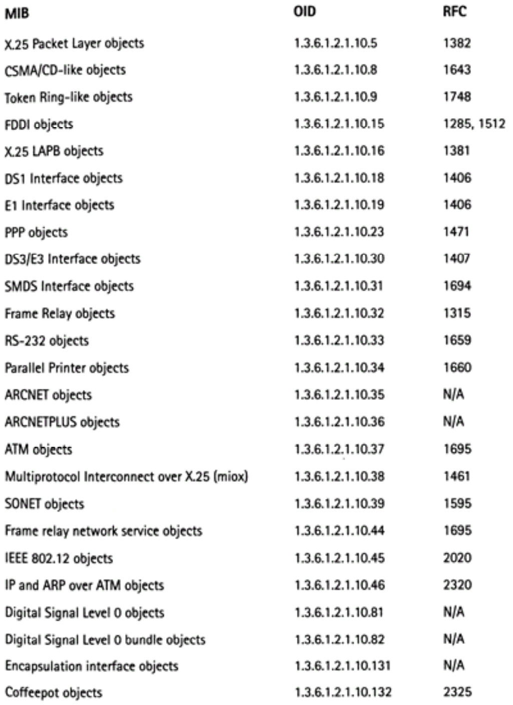
50

# MIBs and RFCs (cont.)

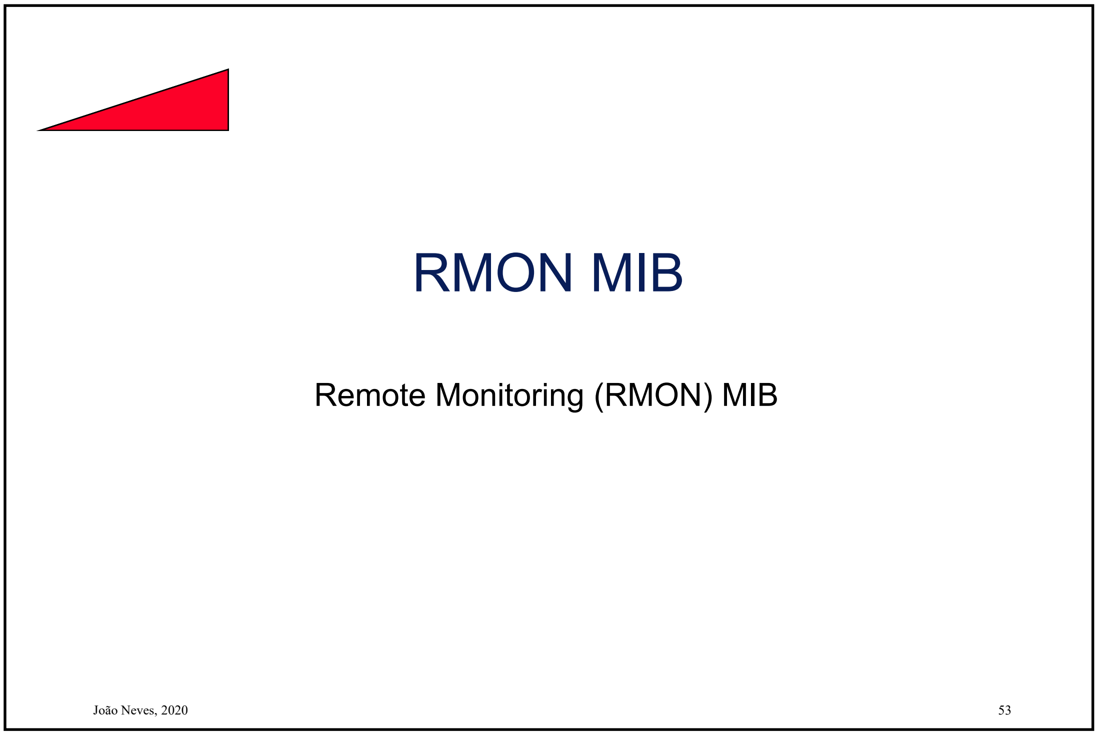


| MIB | OID | RFC |
|---|---|---|
| Relational database objects | 1.3.6.1.2.1.39 | 1697 |
| Traffic flow objects | 1.3.6.1.2.1.40 | 2064 |
| SNA SDLC | 1.3.6.1.2.1.41 | 1747 |
| Token Ring Station Source Route | 1.3.6.1.2.1.42 | 1748 |
| Printer | 1.3.6.1.2.1.43 | 1759 |
| Mobile IP | 1.3.6.1.2.1.44 | 2006 |
| IEEE 802.12 | 1.3.6.1.2.1.45 | 2020 |
| Data Link Switch | 1.3.6.1.2.1.46 | 2024 |
| Entity | 1.3.6.1.2.1.47 | 2037 |
| Internet Protocol | 1.3.6.1.2.1.48 | 2011 |
| Transmission Control Protocol | 1.3.6.1.2.1.49 | 2012 |
| User Datagram Protocol | 1.3.6.1.2.1.50 | 2013 |
| Resource Reservation Protocol | 1.3.6.1.2.1.51 | 2206 |
| Integrated Services | 1.3.6.1.2.1.52 | 2213 |
| IEEE 802.12 Repeater | 1.3.6.1.2.1.53 | 2266 |
| System Application | 1.3.6.1.2.1.54 | 2287 |
| Internet Protocol version 6 | 1.3.6.1.2.1.55 | N/A |
| Internet Control Message Protocol version 6 | 1.3.6.1.2.1.56 | N/A |
| Multicast Address Resolution | 1.3.6.1.2.1.57 | 2417 |

# Transmission Media MIBs

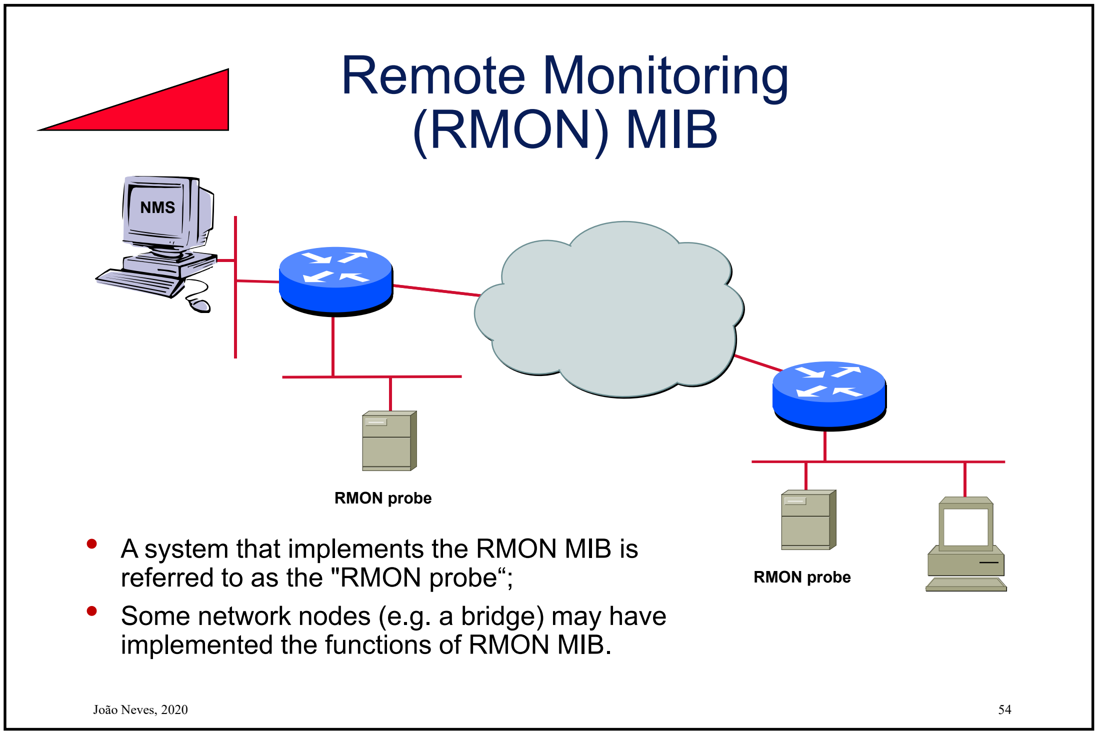


| MIB | OID | RFC |
|---|---|---|
| X.25 Packet Layer objects | 1.3.6.1.2.1.10.5 | 1382 |
| CSMA/CD-like objects | 1.3.6.1.2.1.10.8 | 1643 |
| Token Ring-like objects | 1.3.6.1.2.1.10.9 | 1748 |
| FDDI objects | 1.3.6.1.2.1.10.15 | 1285, 1512 |
| X.25 LAPB objects | 1.3.6.1.2.1.10.16 | 1381 |
| DS1 Interface objects | 1.3.6.1.2.1.10.18 | 1406 |
| E1 Interface objects | 1.3.6.1.2.1.10.19 | 1406 |
| PPP objects | 1.3.6.1.2.1.10.23 | 1471 |
| DS3/E3 Interface objects | 1.3.6.1.2.1.10.30 | 1407 |
| SMDS Interface objects | 1.3.6.1.2.1.10.31 | 1694 |
| Frame Relay objects | 1.3.6.1.2.1.10.32 | 1315 |
| RS-232 objects | 1.3.6.1.2.1.10.33 | 1659 |
| Parallel Printer objects | 1.3.6.1.2.1.10.34 | 1660 |
| ARCNET objects | 1.3.6.1.2.1.10.35 | N/A |
| ARCNETPLUS objects | 1.3.6.1.2.1.10.36 | N/A |
| ATM objects | 1.3.6.1.2.1.10.37 | 1695 |
| Multiprotocol Interconnect over X.25 (miox) | 1.3.6.1.2.1.10.38 | 1461 |
| SONET objects | 1.3.6.1.2.1.10.39 | 1595 |
| Frame relay network service objects | 1.3.6.1.2.1.10.44 | 1695 |
| IEEE 802.12 objects | 1.3.6.1.2.1.10.45 | 2020 |
| IP and ARP over ATM objects | 1.3.6.1.2.1.10.46 | 2320 |
| Digital Signal Level 0 objects | 1.3.6.1.2.1.10.81 | N/A |
| Digital Signal Level 0 bundle objects | 1.3.6.1.2.1.10.82 | N/A |
| Encapsulation interface objects | 1.3.6.1.2.1.10.131 | N/A |
| Coffeepot objects | 1.3.6.1.2.1.10.132 | 2325 |



# RMON MIB

Remote Monitoring (RMON) MIB



# Remote Monitoring (RMON) MIB

- A system that implements the RMON MIB is referred to as the "RMON probe“;
- Some network nodes (e.g. a bridge) may have implemented the functions of RMON MIB.

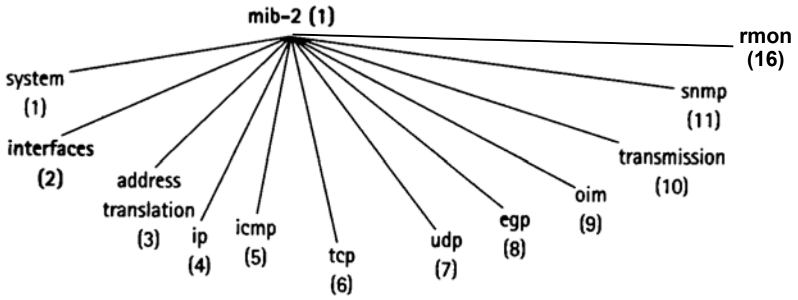
# RMON Goals

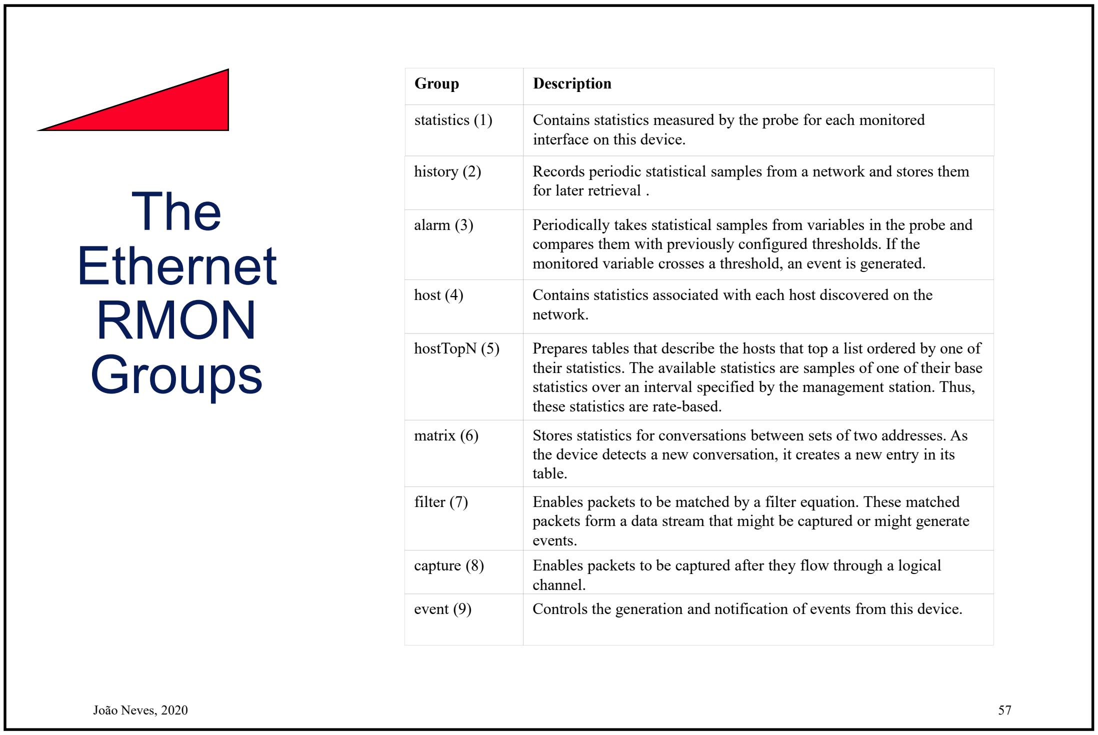
## STD0059, RFC2819

- *Off-line operation* – solve the problem of polling
- *Proactive monitoring* – the monitor can constantly monitor and record the network if it has sufficient resources
- *Problem detection and reporting*
- *Value-added-data* – the monitor analyzes information collected on its subnet, thereby freeing the NMS
- *Multiple managers* – the network may have more than one NMS, then the monitor may be configured to concurrently dialog with the different NMS

# OID of RMON MIB

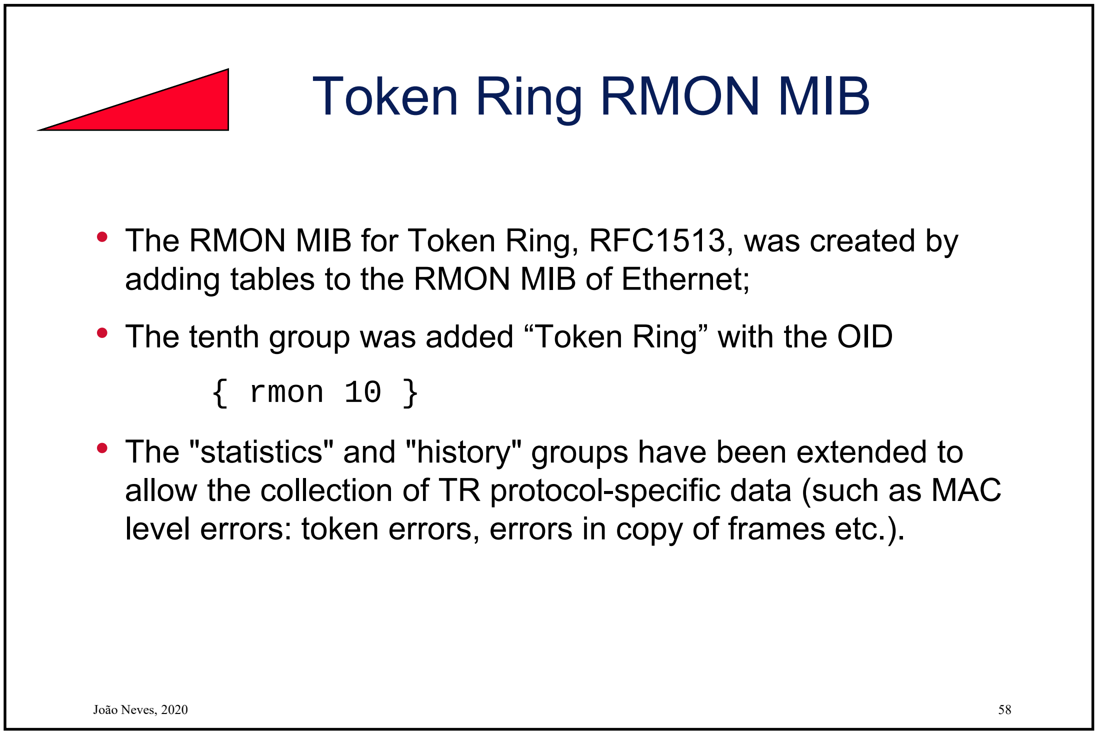
`{1.3.6.1.2.1.16}`





## The Ethernet RMON Groups

| Group | Description |
|---|---|
| statistics (1) | Contains statistics measured by the probe for each monitored interface on this device. |
| history (2) | Records periodic statistical samples from a network and stores them for later retrieval . |
| alarm (3) | Periodically takes statistical samples from variables in the probe and compares them with previously configured thresholds. If the monitored variable crosses a threshold, an event is generated. |
| host (4) | Contains statistics associated with each host discovered on the network. |
| hostTopN (5) | Prepares tables that describe the hosts that top a list ordered by one of their statistics. The available statistics are samples of one of their base statistics over an interval specified by the management station. Thus, these statistics are rate-based. |
| matrix (6) | Stores statistics for conversations between sets of two addresses. As the device detects a new conversation, it creates a new entry in its table. |
| filter (7) | Enables packets to be matched by a filter equation. These matched packets form a data stream that might be captured or might generate events. |
| capture (8) | Enables packets to be captured after they flow through a logical channel. |
| event (9) | Controls the generation and notification of events from this device. |



## Token Ring RMON MIB

- The RMON MIB for Token Ring, RFC1513, was created by adding tables to the RMON MIB of Ethernet;
- The tenth group was added “Token Ring” with the OID

```
{ rmon 10 }
```

- The "statistics" and "history" groups have been extended to allow the collection of TR protocol-specific data (such as MAC level errors: token errors, errors in copy of frames etc.).

# O Grupo tokenRing

- OID

  `{ 1.3.6.1.2.1.16.10 }`

- Contem quatro sub-grupos
  - Ring Station Group – monitoriza os eventos da estação e do anel, tem duas tabelas: a `ringStationControlTable` e `ringStationTable`
  - Ring Station Order Group – mantém a topologia da rede, dá a sequencia das estações nos anéis monitorizados, tem a tabela `ringStationOrderTable`
  - Ring Station Configuration Group – controla a remoção ou a configuração de uma estação do anel, tem duas tabelas: a `ringStationConfigControlTable` e a `ringStationConfigTable`
  - Ring Source Routing Group – contem as estatísticas de utilização do SRB, tem a tabela `sourceRoutingStatsTable`

# RMON2

- The RMON MIBs for Ethernet networks and TR were directed to the management and operation of the layers Physical and Data Link of remote networks;
- RMON2, defined in RFC2021, extends the capabilities of RMON MIB to the upper layers, to the Application layer, adding ten new groups

  `{ rmon 11 }` to `{ rmon 20 }`

- Protocols such as TCP / IP or SPX / IPX can be monitored.

# SMON

Remote Network Monitoring MIB Extensions for Switched Networks Version 1.0 , RFC2613

- The SMON MIB extends RMON MIB allowing the analysis of switched networks.

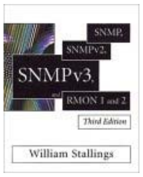
# Comparison

| | MIB-II | HOST | REPEATER | BRIDGE | RMON |
|---|---|---|---|---|---|
| **INTERFACE STATISTICS** | X | | | | |
| **IP, TCP & UDP STATISTICS** | X | | | | |
| **SNMP STATISTICS** | X | | | | |
| **HOST JOB COUNTS** | | X | | | |
| **HOST FILE SYSTEM INFORMATION** | | X | | | |
| **LINK TESTING** | | | X | X | |
| **NETWORK TRAFFIC STATISTICS** | | | X | X | X |
| **TABLE WITH ALL MAC ADDRESSES** | | | X | | X |
| **STATISTICS PER HOST** | | | X | | X |

# Comparison (cont.)

| | MIB-II | HOST | REPEATER | BRIDGE | RMON |
|---|---|---|---|---|---|
| HISTORICAL STATISTICS | | | | | X |
| SPANNING TREE PERFORMANCE | | | | X | |
| WIDE AREA LINK PERFORMANCE | | | | X | |
| TRESHOLDS FOR ANY VARIABLE | | | | | X |
| CONFIGURABLE STATISTICS | | | | | X |
| TRAFFIC MATRIX WITH ALL NODES | | | | | X |
| ‘HOST TOP N’ INFORMATION | | | | | X |
| PACKET / PROTOCOL ANALYSIS | | | | | X |
| DISTRIBUTED LOGGING | | | | | X |

# Bibliography



- **Stallings, William**
  ***SNMP, SNMPv2, SNMPv3 and RMON 1 and 2***
  Addison-Wesley Publishing Company, 3rd Ed.
  ISBN 0-20-148534-6