

# Gestión

Area de Ingeniería Telemática  
<http://www.tlm.unavarra.es>

Grado en Ingeniería en Tecnologías de  
Telecomunicación, 4º

# Modelo para la información

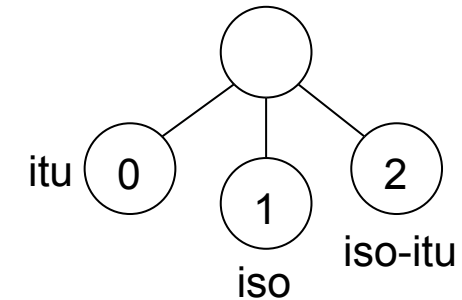
# Modelo para la información

- **SMI** (Structure of Management Information): los mecanismos para describir y nombrar los objetos
- **MIB** (Management Information Base): Es donde se almacena (los objetos) esa información (en agente o gestor), el *schema*
- **MDB** (Management Database) es la base de datos con los datos medidos o introducidos por administración
- Ejemplo:
  - La SMI define cómo especificar un número de puertos de un puente
  - En la MIB dice que un switch del model X del fabricante Y tiene un parámetro que es el número de puertos.
  - En la MDB dice que el switch Z del modelo X del fabricante Y que se está gestionando en la red tiene 12 puertos
  - Veremos casos concretos



# MIB

- *Virtual information store* (el *schema*)
- Estructura en árbol
- Especificada por ISO pero empleada también en Internet
- Objetos definidos empleando ASN.1
  - *Abstract Syntax Notation One*
  - Es una notación formal estándar (ITU-T X.680) para describir datos transmitidos
- Cada objeto tiene: nombre, sintaxis y codificación
- Nombre:
  - OBJECT IDENTIFIER (OID), secuencia de números de nodos de un árbol
  - Raíz sin etiqueta, sub-árboles delegados en gestión

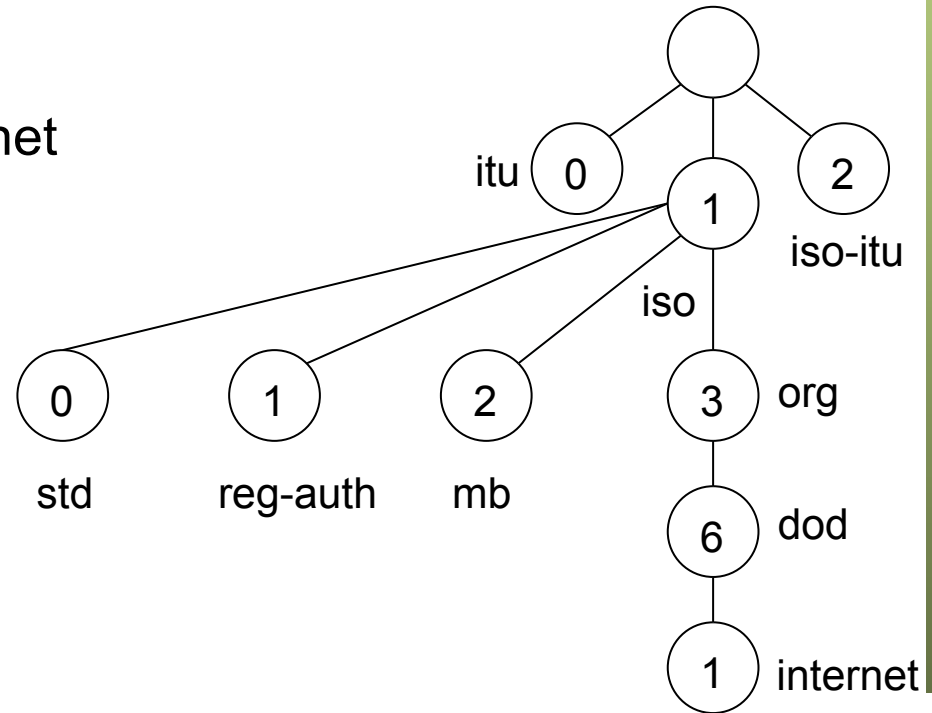


Nota: Podéis encontrar ccitt en lugar de itu



# MIB

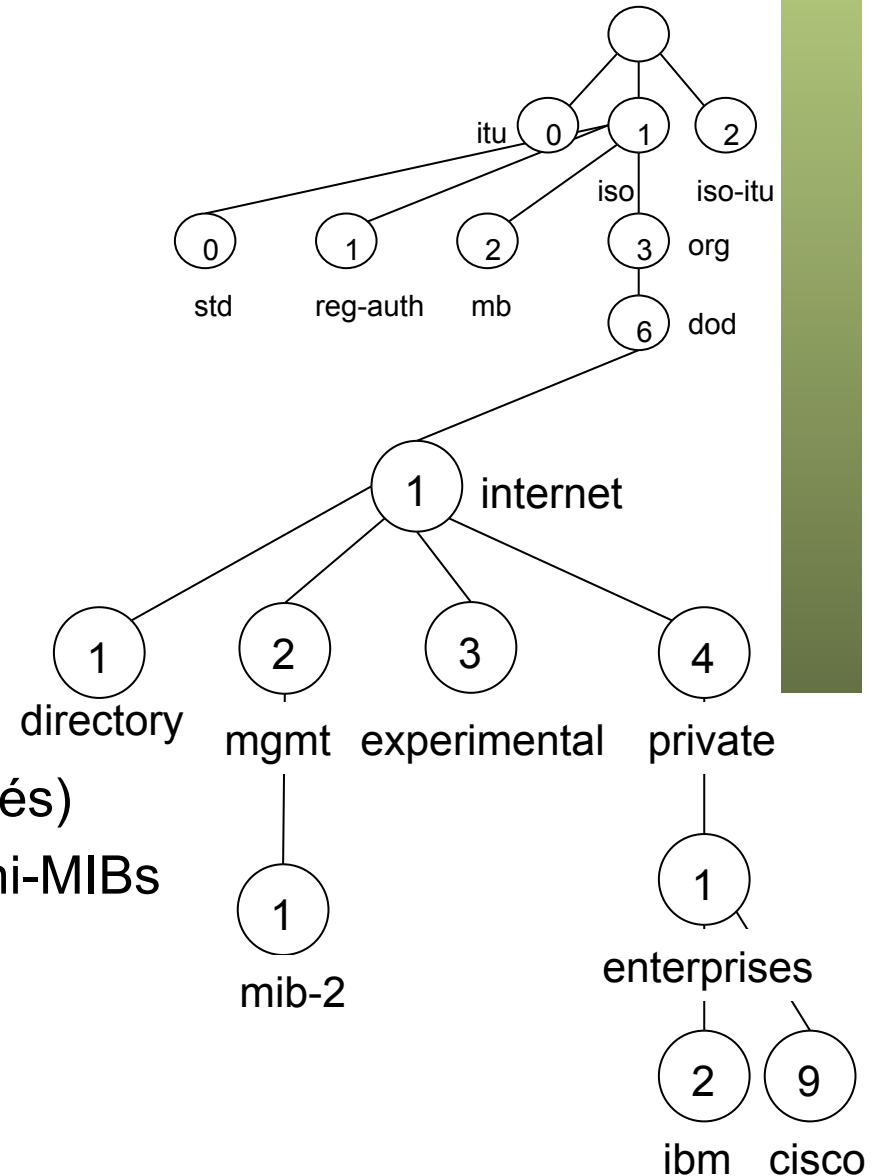
- IANA gestiona:
  - 1.3.6.1 = iso.org.dod.internet
- (...)



# Gestión en Internet

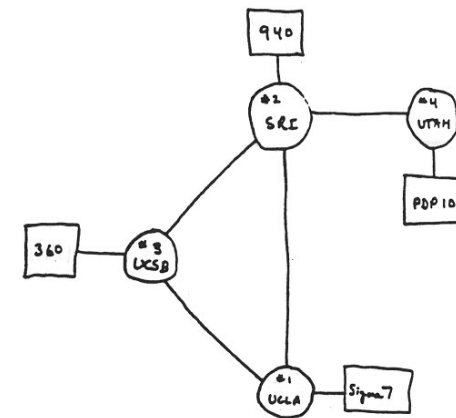
# MIB OIDs para Internet

- IANA gestiona:
  - 1.3.6.1 = iso.org.dod.internet
- 1.3.6.1.4 = private
  - Definidos unilateralmente
  - 1.3.6.1.4.1 = enterprises
  - Ahí un subárbol por empresa
- 1.3.6.1.2 = mgmt
  - Objetos estándar
- 1.3.6.1.2.1 = MIB-II
  - RFC 1213 (actualizada después)
  - Más una gran cantidad de mini-MIBs
  - MIB-I histórica (RFC1156)



# Gestión en Internet

- *Internet Standard Management Framework*
- Última versión (v.3) RFC 3410 (podéis leer ahí sobre su evolución desde v.1)
- *SNMPv3 Management Framework*
- Como con gran parte de Internet, al final su simplicidad es parte del motivo de su éxito (la S es de Simple)
- Simple para no dedicar mucho esfuerzo dado que, total, TCP/IP se acabaría cambiando por los protocolos OSI
- Claro claro...



THE ARPA NETWORK

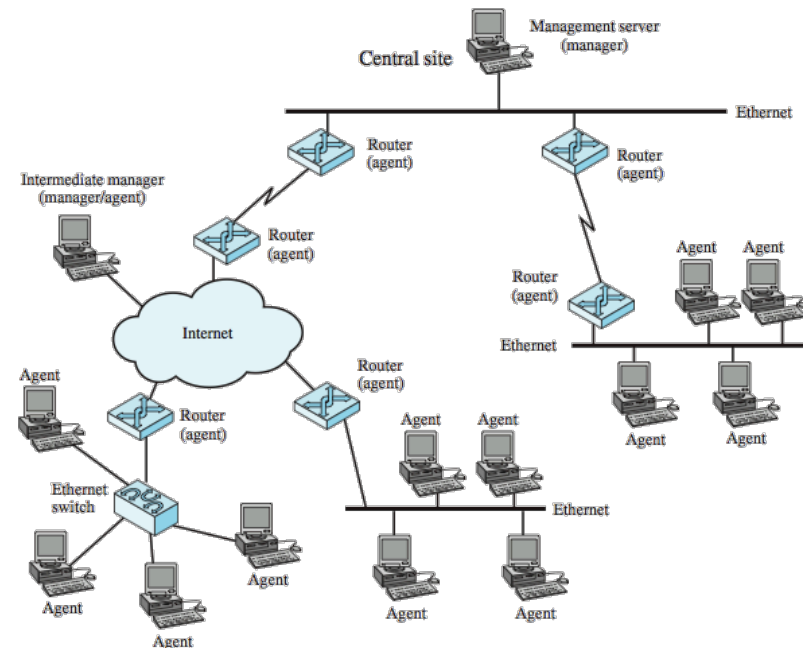
DEC 1969

4 NODES



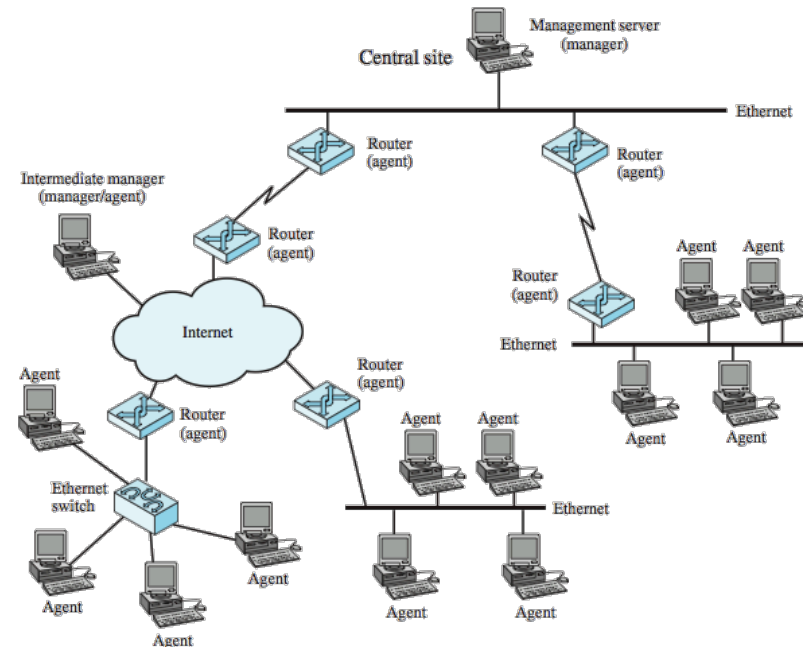
# Gestión en Internet

- Hoy en día la gran mayoría de equipos de red implementan alguna versión de SNMP
- También ha influido que los estándares han sido siempre gratuitos y accesibles por www/ftp
- Pero no le quitamos importancia a la simplicidad
- ¿CMOT? (CMIP over TCP/IP) para una época de transición que aún no ha llegado ;-)



# Gestión en Internet

- El framework de gestión consiste en:
  - Un lenguaje de definición de datos
  - La MIB
  - Un protocolo
  - Seguridad y administración



# SMIv2

- El lenguaje para definir los tipos de datos
- RFC 2578 “Structure of Management Information Version 2 (SMIv2)”
- Es un subconjunto de ASN.1
  - *Abstract Syntax Notation One*
  - Estándar OSI e ITU-T
- SMIv2 es la forma actual de escribir MIBs
- SMIv1 no ha dejado de ser estándar por todos los que dependen de él

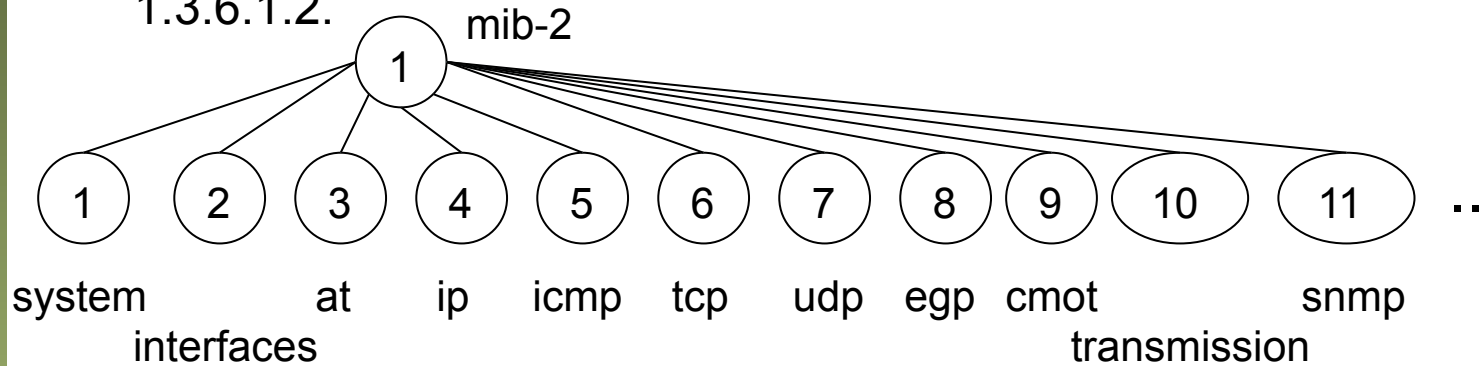
# MIB-II

# MIB-II

- RFC 1213 “Management Information Base for Network Management of TCP/IP-based internets: MIB-II” (año 1991)
  - Grupos: System, Interfaces, Address Translation (IP a MAC), IP, ICMP, TCP, UDP, EGP, Transmission, SNMP

(iso.org.dod.internet.mgmt.)

1.3.6.1.2.



# Ejemplos MIB-II (RFC 1213)

ifOutOctets OBJECT-TYPE

SYNTAX Counter

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The total number of octets transmitted out of the interface, including framing characters."

::= { ifEntry 16 }

ipDefaultTTL OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-write

STATUS mandatory

DESCRIPTION

"The default value inserted into the Time-To-Live field of the IP header of datagrams originated at this entity, whenever a TTL value is not supplied by the transport layer protocol."

::= { ip 2 }

# MIB-II

- RFC 2011 “SNMPv2 Management Information Base for the Internet Protocol using SMIv2” (año 1996)
  - Actualiza objetos empleando SMIv2
  - Para los grupos IP e ICMP

- Ejemplo:

```
ipInDelivers OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS     read-only
    STATUS           current
    DESCRIPTION
        "The total number of input datagrams successfully
delivered
        to IP user-protocols (including ICMP)."
```

```
 ::= { ip 9 }
```

# MIB-II

- Se actualizan con SMIv2 e independizan (año 1996):
  - RFC 2011 “SNMPv2 Management Information Base for the **Internet Protocol** using SMIv2”
  - RFC 2012 “SNMPv2 Management Information Base for the **Transmission Control Protocol** using SMIv2”
  - RFC 2013 “SNMPv2 Management Information Base for the **User Datagram Protocol** using SMIv2”
- Pero...
- Obsoletas por:
  - RFC 4293 “Management Information Base for the Internet Protocol (IP)” (año 2006)
  - RFC 4022 “Management Information Base for the Transmission Control Protocol (TCP)” (año 2005)
  - RFC 4113 “Management Information Base for the User Datagram Protocol (UDP)” (año 2005)
- Que actualizan objetos, mejoran soporte de IPv6, recogen objetos de otras RFCs, etc.



# Ejemplo

- Objeto mencionado antes, de la RFC 2011 :

```
ipInDelivers OBJECT-TYPE
```

```
SYNTAX Counter32
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

```
"The total number of input datagrams successfully delivered  

to IP user-protocols (including ICMP)."
```

```
::= { ip 9 }
```

- “Deprecated” en la RFC 4293:

```
ipInDelivers OBJECT-TYPE
```

```
SYNTAX Counter32
```

```
MAX-ACCESS read-only
```

```
STATUS deprecated
```

```
DESCRIPTION
```

```
"The total number of input datagrams successfully delivered  

to IPv4 user-protocols (including ICMP)."
```

```
This object has been deprecated as a new IP version neutral  

table has been added. It is loosely replaced by  

ipSystemStatsIndelivers."
```

```
::= { ip 9 }
```

# Ejemplo

- Otro ejemplo (RFC 4293)

```
ipSystemStatsInBcastPkts OBJECT-TYPE
```

```
SYNTAX Counter32
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

```
    "The number of IP broadcast datagrams received.  
    Discontinuities in the value of this counter can occur at  
    re-initialization of the management system, and at other  
    times as indicated by the value of  
    ipSystemStatsDiscontinuityTime."
```

```
::= { ipSystemStatsEntry 42 }
```

# MIBs

- RFC 1213, grupos:
  - system: generalidades sobre el elemento
  - interfaces: características, parámetros y contadores de interfaces
  - at (address translation): tabla ARP pero deprecated en MIB-II y se mueve al grupo ip
  - ip, icmp, tcp, udp: movidos a RFCs independientes
  - egp
  - snmp: contadores de mensajes y errores
- Otras:
  - appletalk (RFC 1742)
  - ospf (última versión RFC 4750)
  - bgp (RFC 4273)
  - rmon, dot1dBridge, snmpDot3RptMgt, rip2, host, snmpDot3MauMgt, ifMIB, snanauMIB, etherMIB, atmMIB, snaDLC, mipMIB, dlsr, entityMIB, ipMIB (RFC 4293), tcpMIB (RFC 4022), udpMIB, rsvp, intSrc, sysAppMIB, ipv6MIB, radiusMIB, vrrpMIB, docsDev, nhrpMIB, fcFeMIB, inetAddressMIB, ptopoMIB, ipMRouteStdMIB, ianaRtProtoMIB, igmpStdMIB, ...

# Otras MIBs

- RFC 2020 “IEEE 802.12 Interface MIB”
- RFC 2789 “Mail Monitoring MIB”
- RFC 4292 “IP Forwarding Table MIB”
- RFC 4750 “OSPF Version 2 Management Information Base”
- RFC 4780 “Management Information Base for the Session Initiation Protocol (SIP)”
- RFC 4802 “Generalized Multiprotocol Label Switching (GMPLS) Traffic Engineering Management Information Base”
- RFC 4803 “Generalized Multiprotocol Label Switching (GMPLS) Label Switching Router (LSR) Management Information Base”
- RFC 4805 “Definitions of Managed Objects for the DS1, J1, E1, DS2, and E2 Interface Types”
- RFC 5060 “Protocol Independent Multicast MIB”
- RFC 5066 “Ethernet in the First Mile Copper (EFMCu) Interfaces MIB”
- RFC 5650 “Definitions of Managed Objects for Very High Speed Digital Subscriber Line 2 (VDSL2)”
- Buscando “MIB” en los títulos de RFCs salen 376 resultados...

etc, etc, etc, etc, etc, etc, etc  
etc, etc, etc, etc, etc, etc, etc  
etc, etc, etc, etc, etc, etc, etc  
etc, etc, etc, etc, etc, etc, etc

# MIB: Sintaxis

# Sintaxis: OID

- Como decíamos, un subconjunto de capacidades de ASN.1
- El cual está basado en BNF (*Backus-Nauer Form*), que define así:

`<name> ::= <definition>`

- Con un ejemplo:

```
ipSystemStatsInBcastPkts OBJECT-TYPE
```

```
SYNTAX Counter32
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

```
"The number of IP broadcast datagrams received.  

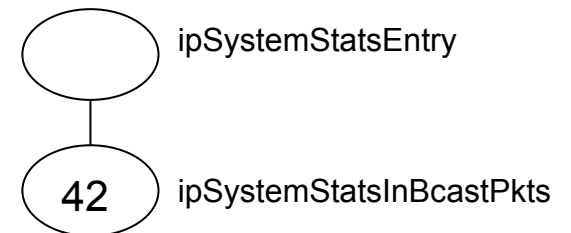
Discontinuities in the value of this counter can occur at  

re-initialization of the management system, and at other  

times as indicated by the value of  

ipSystemStatsDiscontinuityTime."
```

```
::= { ipSystemStatsEntry 42 }
```



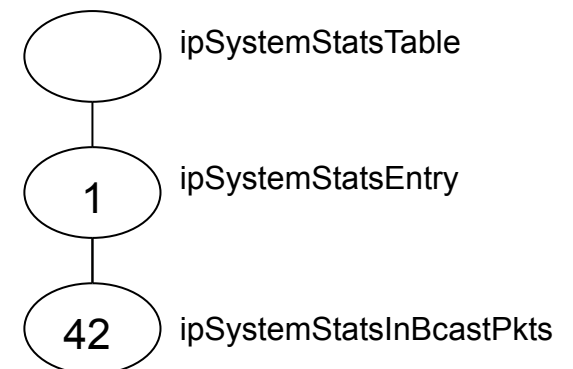
# Sintaxis: OID

- Como decíamos, un subconjunto de capacidades de ASN.1
- El cual está basado en BNF (*Backus-Naur Form*), que define así:

`<name> ::= <definition>`

- Si la buscamos:

```
ipSystemStatsEntry OBJECT-TYPE
    SYNTAX      IpSystemStatsEntry
    MAX-ACCESS not-accessible
    STATUS      current
    DESCRIPTION
        "A statistics entry containing system-wide objects for a
        particular IP version."
    INDEX { ipSystemStatsIPVersion }
    ::= { ipSystemStatsTable 1 }
```



# Sintaxis: OID

- Como decíamos, un subconjunto de capacidades de ASN.1
- El cual está basado en BNF (*Backus-Nauer Form*), que define así:

`<name> ::= <definition>`

- Si la buscamos:

`ipSystemStatsTable OBJECT-TYPE`

`SYNTAX SEQUENCE OF IpSystemStatsEntry`

`MAX-ACCESS not-accessible`

`STATUS current`

`DESCRIPTION`

"The table containing system wide, IP version specific traffic statistics. This table and the ipIfStatsTable contain similar objects whose difference is granularity. Where this table contains system wide traffic statistics, the ipIfStatsTable contains the same statistics but counted on a per-interface basis."

`::= { ipTrafficStats 1 }`





# Sintaxis: OID

- Como decíamos, un subconjunto de capacidades de ASN.1
- El cual está basado en BNF (*Backus-Nauer Form*), que define así:

`<name> ::= <definition>`

- Si la buscamos:

`ipTrafficStats OBJECT IDENTIFIER ::= { ip 31 }`

- Es decir:

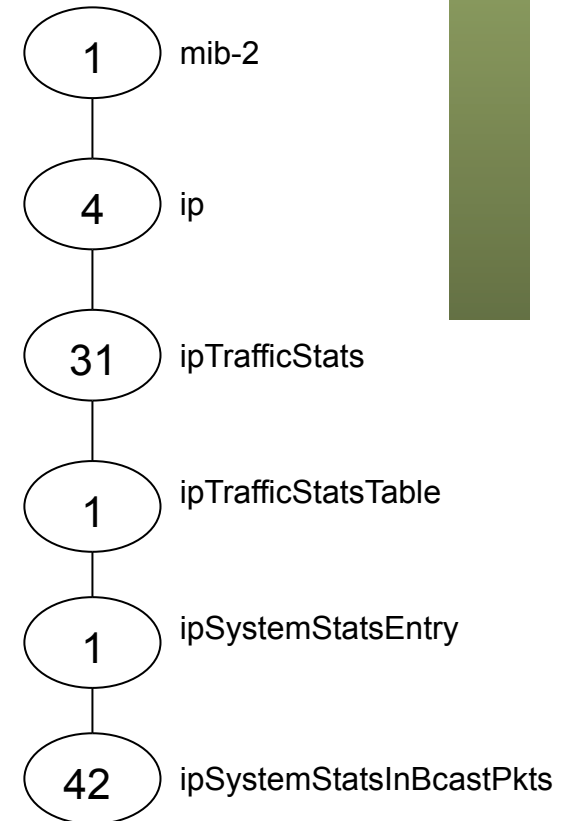
`iso.org.dod.internet.mgmt.mib-2.ip.ipTrafficStats.  
 ipTrafficStatsTable.ipSystemStatsEntry.ipSystemSta  
 tsInBcastPkts`

- O más claro:

`1.3.6.1.2.1.4.31.1.1.42`

- Eso hace referencia al tipo de objeto, la instancia se accede con .0 al final, salvo en tablas

“Aragorn II, hijo de Arathorn II, hijo de Arador, hijo de Argonui, hijo de Arathorn I, hijo de Arassuil, hijo de Arahad II, hijo de Aravorn, hijo de Aragorst, hijo de Arahad I, hijo de Araglas, hijo de Aragorn, hijo de Aravir, hijo de Aranuir, hijo de Arahael, hijo de Aranarth, hijo de Arvedui, hijo de Araphant, hijo de Araval, hijo de Argeleb II... heredero de Elendil”



# Sintaxis

- OBJECT-TYPE permite definir un tipo de objeto gestionado
- SYNTAX (obligatorio)
  - Define el tipo
  - INTEGER, Integer32, OCTET STRING, OBJECT IDENTIFIER, BITS, IpAddress, Counter32, Gauge32, TimeTicks, Opaque, Counter64, Unsigned32
  - Puede ser una tabla con SEQUENCE OF <tipo>
  - “Counter” no se puede decrementar y vuelve al comienzo al desbordarse
  - “Gauge” no se sale del máximo o mínimo
  - Direcciones IPv6 como texto en un tipo OCTET STRING (SIZE (0..8))

ipSystemStatsInBcastPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP broadcast datagrams received.  
 Discontinuities in the value of this counter can occur at  
 re-initialization of the management system, and at other  
 times as indicated by the value of  
 ipSystemStatsDiscontinuityTime."

::= { ipSystemStatsEntry 42 }

# Sintaxis

- MAX-ACCESS (obligatorio)
  - Define si tiene sentido leer, crear o modificar instancias del objeto
  - También si se puede emplear en notificaciones
  - Valores (de mayor a menor): read-create, read-write, read-only, accessible-for-notify, not-accessible

```
ipSystemStatsInBcastPkts OBJECT-TYPE
```

```
SYNTAX Counter32
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

```
    "The number of IP broadcast datagrams received.  

    Discontinuities in the value of this counter can occur at  

    re-initialization of the management system, and at other  

    times as indicated by the value of  

    ipSystemStatsDiscontinuityTime."
```

```
::= { ipSystemStatsEntry 42 }
```

# Sintaxis

- STATUS (obligatorio)
  - “current” : ok
  - “obsolete” : no debería implementarse y puede eliminarse si ya se implementa
  - “deprecated” : obsoleta pero permite implementarla por interoperatividad con implementaciones existentes

ipSystemStatsInBcastPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of IP broadcast datagrams received.  
 Discontinuities in the value of this counter can occur at  
 re-initialization of the management system, and at other  
 times as indicated by the value of  
 ipSystemStatsDiscontinuityTime."

::= { ipSystemStatsEntry 42 }

# Sintaxis

- DESCRIPTION (obligatorio)
  - Definición en texto del objeto
  - Incluye toda la semántica del mismo
- Hay otros opcionales (ver RFC 2578)

```
ipSystemStatsInBcastPkts OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
```

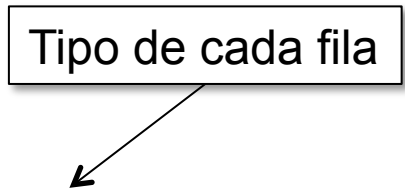
## DESCRIPTION

```
"The number of IP broadcast datagrams received.
Discontinuities in the value of this counter can occur at
re-initialization of the management system, and at other
times as indicated by the value of
ipSystemStatsDiscontinuityTime."
```

```
::= { ipSystemStatsEntry 42 }
```

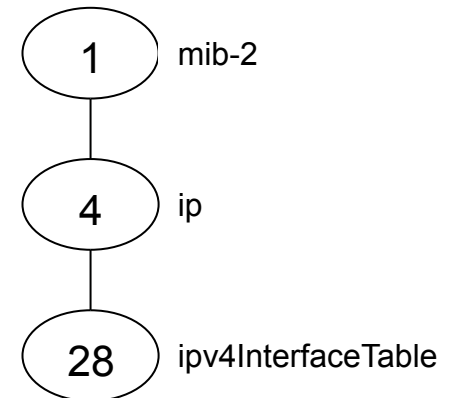
# Tablas

- Se definen como hemos dicho con SEQUENCE OF
- Es una tabla conceptual y por definición no accesible
- Ejemplo (RFC 4293):



```

ipv4InterfaceTable OBJECT-TYPE
  SYNTAX      SEQUENCE OF Ipv4InterfaceEntry
  MAX-ACCESS  not-accessible
  STATUS      current
  DESCRIPTION
    "The table containing per-interface IPv4-specific
    information."
  ::= { ip 28 }
  
```



# Tablas

- Cada fila es de tipo SEQUENCE con los elementos de cada fila
- Ejemplo (RFC 4293):

```
Ipv4InterfaceEntry ::= SEQUENCE {  
    ipv4InterfaceIfIndex          InterfaceIndex,  
    ipv4InterfaceReasmMaxSize    Integer32,  
    ipv4InterfaceEnableStatus    INTEGER,  
    ipv4InterfaceRetransmitTime  Unsigned32  
}
```

# Tablas

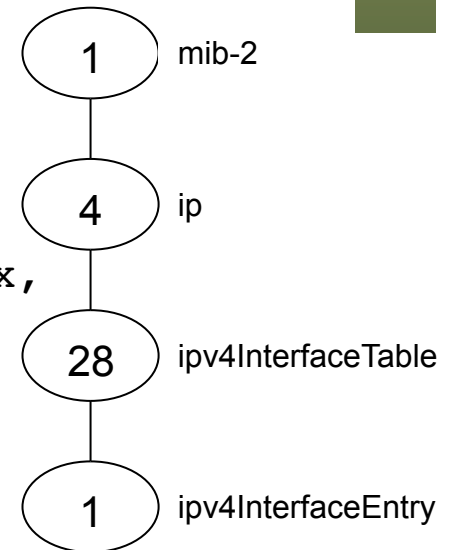
- Tenemos otro objeto por debajo de la tabla del mismo tipo que la secuencia
- INDEX: determina qué objeto(s) de la fila permite(n) distinguirla
- Ejemplo (RFC 4293):

```

ipv4InterfaceEntry OBJECT-TYPE
    SYNTAX      Ipv4InterfaceEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry containing IPv4-specific information for a specific
        interface."
    INDEX { ipv4InterfaceIfIndex }
    ::= { ipv4InterfaceTable 1 }
  
```

```

Ipv4InterfaceEntry ::= SEQUENCE {
    ipv4InterfaceIfIndex      InterfaceIndex,
    ipv4InterfaceReasmMaxSize  Integer32,
    ipv4InterfaceEnableStatus INTEGER,
    ipv4InterfaceRetransmitTime Unsigned32
}
  
```





# Tablas

- Cada elemento de la secuencia tiene su identificador
- Ejemplo (RFC 4293):

```

Ipv4InterfaceEntry ::= SEQUENCE {
    ipv4InterfaceIfIndex      InterfaceIndex,
    ipv4InterfaceReasmMaxSize Integer32,
    ipv4InterfaceEnableStatus INTEGER,
    ipv4InterfaceRetransmitTime Unsigned32
}
  
```

**ipv4InterfaceEnableStatus** OBJECT-TYPE

```

SYNTAX      INTEGER {
                up(1),
                down(2)
            }
  
```

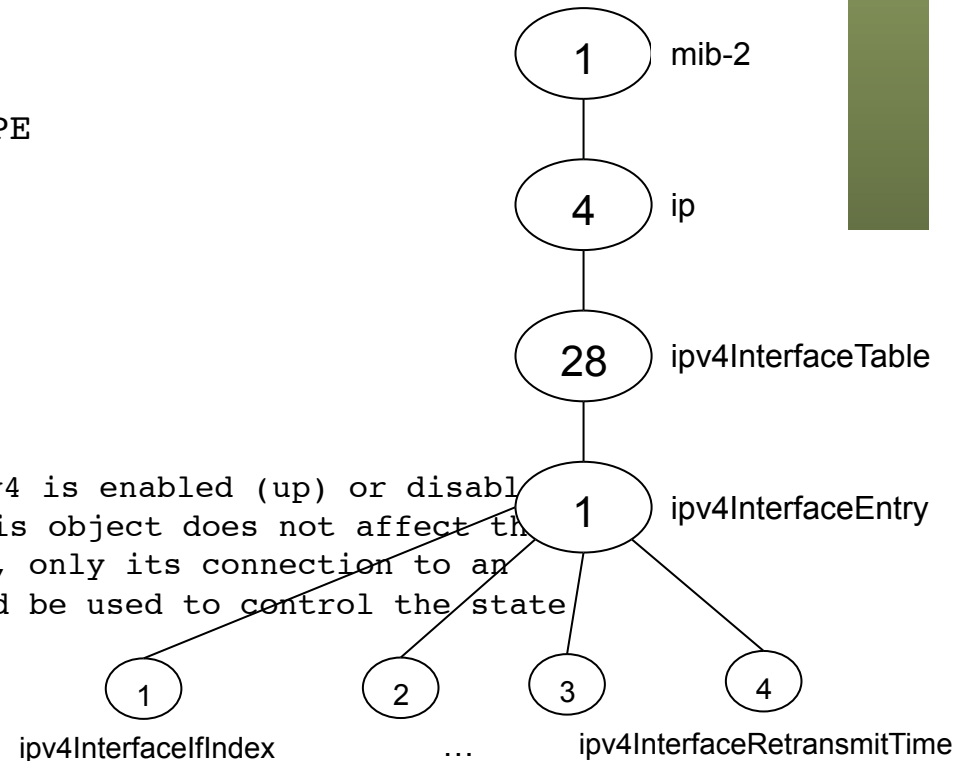
MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The indication of whether IPv4 is enabled (up) or disabled (down) on this interface. This object does not affect the state of the interface itself, only its connection to an IPv4 stack. The IF-MIB should be used to control the state of the interface."

```
 ::= { ipv4InterfaceEntry 3 }
```



# Tablas: ejemplo acceso a datos

- La columna ipv4InterfaceEnableStatus es  
`ip.ipv4InterfaceTable.ipv4InterfaceEntry.ipv4EnableStatus`
- o desde la raíz

`1.3.6.1.2.1.4.28.1.3`

- Para especificar una fila en concreto hay que añadir el índice que en este caso es justo el valor del ipv4InterfaceIndex
- Así pues

`1.3.6.1.2.1.4.28.1.3.3`

daría 2 (down)

- El índice puede estar formado por varias columnas

<code>ipv4InterfaceIndex</code>	<code>ipv4InterfaceResmMaxSize</code>	<code>ipv4InterfaceEnableStatus</code>	<code>ipv4InterfaceRetransmitTime</code>
1	65536	1 (up)	1000
2	65536	1 (up)	1000
3	32768	2 (down)	2000
4	8192	1 (up)	1000

# Tablas: creación de filas

- Debe haber una columna con *SYNTAX* de *RowStatus* y *MAX-ACCESS* de *read-create*
- El procedimiento es complejo, con dos modos para hacerlo, la fila puede pasar por varios estados y requerir varios comandos para completarse

# MIB: Sintaxis

# Ejemplos de MIBs

- MIB-II

<http://www.rfc-editor.org/rfc/rfc1213.txt>

- IP, ICMP

<http://www.rfc-editor.org/rfc/rfc4293.txt>

- TCP

<http://www.rfc-editor.org/rfc/rfc4022.txt>

- UDP

<http://www.rfc-editor.org/rfc/rfc4113.txt>

- BGP

<http://www.rfc-editor.org/rfc/rfc4273.txt>

- MPLS LSR

<http://www.rfc-editor.org/rfc/rfc3813.txt>

- Algunas webs para navegar MIBs

<http://www.snmplink.org/OnLineMIB/Standards/>

<http://www.simpleweb.org>

<http://tools.cisco.com/Support/SNMP/do/BrowseOID.do>