Gestió de xarxa

Pere Barberán Agut

P07/19017/02823

UOC
www.uoc.edu

Índex

Introducció

L'imparable creixement els últims vint-i-cinc anys en les necessitats de processament de la informació ha anat acompanyat del ràpid desenvolupament en els ordinadors i de la tecnologia de xarxa per a suportar aquestes necessitats. Lligat a aquesta demanda hi ha hagut una explosió en la varietat d'equips i xarxes ofertes pels venedors. Queden lluny els dies en què les empreses confiaven en un únic venedor i en conseqüència en una arquitectura que suportés les seves necessitats.

Avui en dia les organitzacions creixen ràpidament i la seva arquitectura de xarxa pot estar formada per diverses LAN, WAN, suportades per commutadors (*switchs*), encaminadors, diferents tipus de computadors, serveis, etc. Per gestionar aquests sistemes i xarxes calen eines i aplicacions que treballin de manera automatitzada. La base d'aquestes eines en un entorn multivenedor és que hi hagi tècniques estandarditzades tant per a representar com per a intercanviar la informació de gestió de xarxes.

Així, en el món d'Internet l'estàndard escollit ha estat *simple network management protocol* (SNMP) i l'especificació relacionada Remote *network monitoring* (RMON). Aquests estàndards inicials han estat revisats i actualitzats amb diverses versions en els últims anys.

Dintre del món de les telecomunicacions el procés ha estat similar quant a creixement. Actualment és molt competitiu, atesa la liberalització i globalització del mercat de les telecomunicacions. Els estàndards TMN estan creixent en importància com un medi per a proporcionar un avantatge als proveïdors de serveis de telecomunicacions.

En aquest mòdul didàctic es vol donar una visió dels conceptes associats a la gestió de xarxa i els protocols més àmpliament acceptats, com són SNMP i TMN.

Objectius

Els materials didàctics d'aquest mòdul han de permetre que l'estudiant assoleixi els objectius següents:

1. Entendre la necessitat de la gestió de xarxa, els seus requeriments i l'arquitectura bàsica de la gestió.

2. Conèixer els elements de la gestió de xarxa SNMP aprofundint en cadascun d'aquests elements i veient els avantatges i les limitacions de les diverses versions del protocol.

3. Aprendre l'estructura de la MIB-2 i ser capaç de fer peticions SNMP als dispositius gestionats.

4. Entendre els objectius del monitoratge remot RMON, com es configura i com s'accedeix als recursos emmagatzemats en les sondes.

5. Diferenciar les necessitats de les operadores de telecomunicacions respecte a les xarxes convencionals.

6. Conèixer l'arquitectura bàsica de gestió TMN per a infraestructures de telecomunicacions.

1. Introducció a la gestió de xarxa

Igual que les xarxes porten informació, aquesta informació necessita ser transmesa de manera eficient, efectiva i fiable. Hi ha moltes raons que fan que calgui una gestió de xarxa:

- Les xarxes actuals són una col·lecció de hardware i software de fabricants diversos i que cal configurar perquè funcionin conjuntament. Cada venedor a la vegada té les seves eines per a gestionar els seus equips de la forma més adequada possible.

- Les noves tecnologies han de conviure amb tecnologies antigues abans que aquestes siguin actualitzades. Cal, per tant, configurar xarxes amb tecnologies de diverses "generacions".

- La gestió de xarxa és un factor de cost. Les empreses intenten que la planificació, el suport, la detecció, la correcció de problemes, l'operació i l'administració siguin tasques que tinguin una component el més automatitzada possible.

- La seguretat és un altre aspecte molt important en les xarxes actuals.

- Altres aspectes relacionats amb la gestió són la planificació i el disseny, la gestió de polítiques, la gestió de sistemes, la gestió de serveis que es fan amb eines normalment propietàries.

Així, el marc de treball de la gestió de xarxa ha d'estar dirigit a resoldre aquests temes dintre dels límits funcionals i de cost. El cost de la gestió de xarxa no es refereix només al preu de compra dels equips, sinó que s'ha de veure des d'un punt de vista més ampli i ha de preveure aspectes humans, actualitzacions, utilització dels sistemes, etc.

Es pot dir, per tant, que la gestió de xarxa optimitza la capacitat funcional de la xarxa. Algunes connotacions que se'n poden derivar són les següents:

- Manté la xarxa operant a rendiment màxim.
- Informa a l'operador de qualsevol anomalia i intenta acotar el lloc on s'ha produït.
- Possibilita examinar el rendiment de la xarxa en temps real.
- Proporciona funcions de gestió de la configuració.

L'objectiu i els beneficis de les plataformes actuals de gestió de xarxa són els següents:

- Millorar en el procés de la presa de decisions.
- Millorar en l'eficàcia de les operacions.
- Alliberar els gestors de les feines rutinàries per poder-se dedicar a l'anàlisi, el control i la planificació. Gran part de les falles solen ser rutinàries.
- Assolir el ràpid retorn de la inversió.

1.1. Fonaments de la gestió de xarxa

Què és la gestió de xarxa?

La ISO defineix la gestió de xarxa com el conjunt d'elements de control i supervisió dels recursos que permeten que la comunicació tingui lloc sobre la xarxa.

1.1.1. Requeriments de la gestió de xarxa

La ISO ha definit les principals àrees funcionals de la gestió de xarxa. Encara que aquesta classificació ha estat desenvolupada per l'OSI, ha obtingut un ampli reconeixement. Les àrees funcionals són les següents

1) Gestió de fallades. Quan hi ha una fallada és molt important que l'administrador pugui fer el següent el més ràpidament possible:

a) Determinar on s'ha produït la fallada.

b) Aïllar la resta de la xarxa de la fallada perquè pugui continuar funcionant sense interferències.

c) Reconfigurar o modificar la xarxa de manera que es minimitzi l'impacte d'operació sense que fallin els diversos components.

d) Reparar o reemplaçar el component perquè la xarxa torni a funcionar de forma correcta.

2) Gestió de rendiment. Es defineix com l'avaluació del comportament dels elements de la xarxa. Per poder fer aquesta anàlisi és necessari mantenir un històric amb dades estadístiques i de configuració.

3) Gestió de comptes. Determinació dels costos associats a l'ús dels recursos i l'assignació de les càrregues corresponents.

4) Gestió de configuració. La gestió de configuració s'encarrega de les tasques següents:

a) Inicialització i desactivació

b) Definició o canvi de paràmetres de configuració

c) Recollida d'informació d'estat

d) Denominació dels elements de la xarxa

5) Gestió de seguretat. Inclou tota una sèrie de facilitats mitjançant les quals l'administrador de xarxa modifica les funcions que proporcionen seguretat davant els intents d'accessos no autoritzats. Hi trobaríem aspectes com la gestió de claus, els tallafocs o els històrics de seguretat.

Abast dels requeriments de la gestió de la xarxa

En la majoria de casos les plataformes actuals no cobreixen tots els requeriments de la gestió de la xarxa.

1.1.2. Components de la gestió de xarxa

La gestió de xarxa està formada bàsicament per quatre components (figura 1):

1) Plataforma de gestió o element que supervisa. Aquest component s'anomena *network management server* (NMS) i s'usa per a gestionar tota la xarxa. Rep tota la informació i la mostra un cop **filtra** el que realment és important.

Funció del gestor de la xarxa

Si el gestor no fes un filtrat de la informació que realment és útil, la gestió de xarxa acabaria no resultant interessant atès l'excés d'informació. Cal, per tant, que el gestor elimini la informació supèrflua.

2) L'element per gestionar. Aquest element s'anomena *network management agent* (NMA). L'NMA és un element de la xarxa com poden ser els encaminadors, commutadors, *hosts*, etc. A la vegada, cada node de xarxa conté una col·lecció de software encarregat de les tasques de gestió. Aquest software s'anomena NME (*network management entity*).

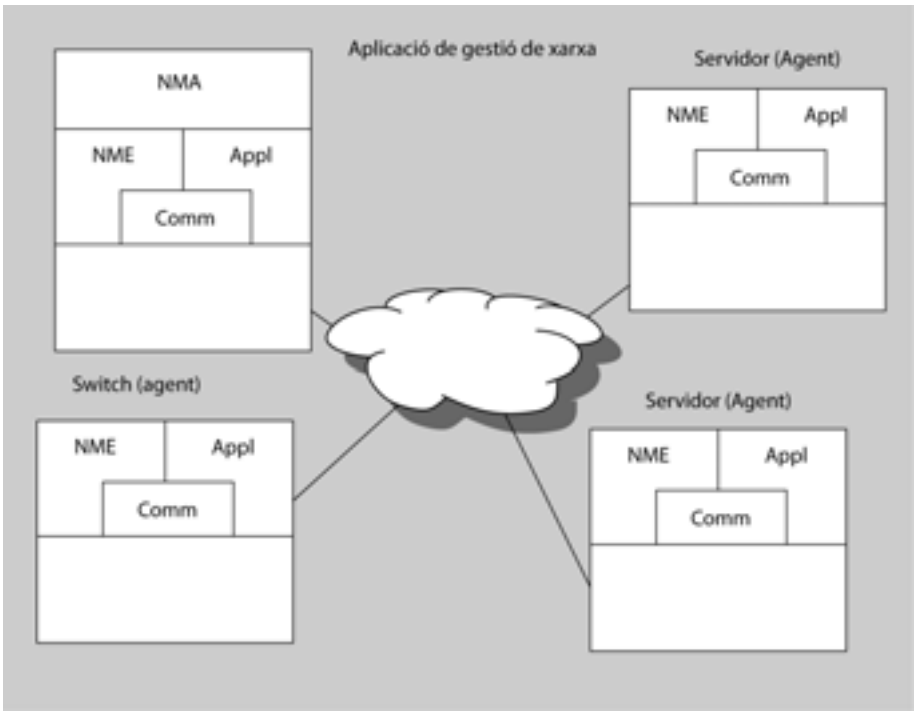
3) El protocol de diàleg entre l'agent i el servidor. El protocol més comú és SNMP (*simple network management protocol*). Genèricament seria el software de comunicació.

Exemples d'objectes per gestionar

Alguns exemples d'objectes per gestionar poden ser les taules d'encaminament d'un encaminador, l'última vegada que es va reiniciar un equip, etc.

4) Els objectes per gestionar. Aquests objectes són els recursos dels dispositius per gestionar. S'agrupen normalment en una mena de base de dades anomenada MIB (*management information base*).

Figura 1. Elements d'un sistema de gestió



2. Gestió de xarxa SNMP

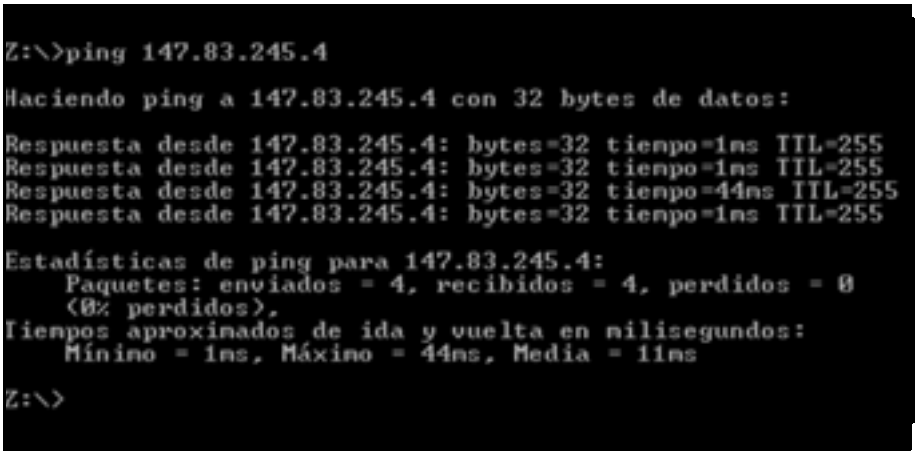
La gestió de xarxa SNMP (*simple network management protocol*) engloba tot un grup d'especificacions de gestió de xarxa que inclouen el **protocol**, la **definició de l'estructura de dades** i altres conceptes associats.

2.1. Orígens de la gestió de xarxa SNMP

Als inicis del que coneixem com Internet no existien els protocols de gestió com a tal. L'única eina que es feia servir per "gestió" era el protocol ICMP (*Internet control message protocol*). ICMP està disponible en tots els dispositius IP i en els seus inicis era un mecanisme que servia per a testejar (*echo/reply*) la comunicació entre parelles de dispositius. El receptor del missatge "echo" està obligat a respondre a qui li fa la petició. A la vegada, també és útil per a obtenir altres informacions molt interessants, com són el retard en la xarxa.

L'exemple més conegut de l'ús del protocol ICMP és el programa **ping** (*packet Internet grouper*). Amb aquesta senzilla eina es poden observar, entre altres, les variacions en el retard extrem a extrem, la pèrdua de paquets, de manera que pot ajudar entre altres coses a determinar àrees de congestió i punts de fallada.

Figura 2. Captura d'una petició ping des de *cmd*



Aquesta eina juntament amb altres eines senzilles van ser solucions prou vàlides per a la gestió de xarxa fins a mitjan anys vuitanta. A partir de llavors, amb el creixement exponencial d'Internet es va fer necessari plantejar noves eines per a la gestió.

Les propostes més interessants van ser:

- *Simple network management protocol* (SNMP)
- CMIP sobre TCP/IP (CMOT)

El 1988 Internet Architecture Board (IAB) va revisar les propostes i va aprovar que SNMP fos desenvolupat com una solució a curt termini mentre que es plantejava CMOT com la solució cap a on s'havia d'anar. La simplicitat del protocol SNMP va fer que aviat fos el protocol escollit pels fabricants.

Fent una breu cronologia de l'evolució d'SNMP tenim el següent:

1) SNMPv1. Sorgeix i introdueix les bases a la gestió de xarxes.

2) SNMPv2. Millora capacitats de transferència de dades i intercanvi d'informació entre gestors. (*getbulkrequest*, *informrequest*) i millora la manera de gestionar les taules.

3) RMONv1. Permet la monitorització remota d'un segment de xarxa.

4) RMONv2. Amplia la MIB-II d'RMON i permet un monitoratge de segments de xarxes LAN i un seguiment del trànsit a nivell de xarxa i aplicació.

5) SNMPv3. Introdueix capacitats de seguretat (control d'accés, autenticació i privacitat).

2.2. Arquitectura de gestió de xarxa SNMP

El model de gestió de xarxa TCP/IP té els elements clau següents:

- L'estació de gestió o gestor
- Els agents gestionats
- La base de dades de gestió (*managament information base*, MIB)
- El protocol de gestió de xarxa

L'**estació de gestió** pot ser o un o un grup de dispositius distribuïts. L'estació de gestió serveix d'interfície entre el gestor i el sistema de gestió de xarxa.

Si us hi fixeu, els elements de la xarxa TCP/IP són els mateixos ja comentats en el subpartat 1.1.2.

Els **agents gestionats** són l'equipament lògic allotjat en els dispositius per gestionar de la xarxa. Emmagatzema dades de gestió i respon a les peticions associades a aquestes dades. De forma asíncrona (*traps*) –dels quals parlarem més endavant– poden proporcionar informació al gestor sense que aquest ho sol·liciti.

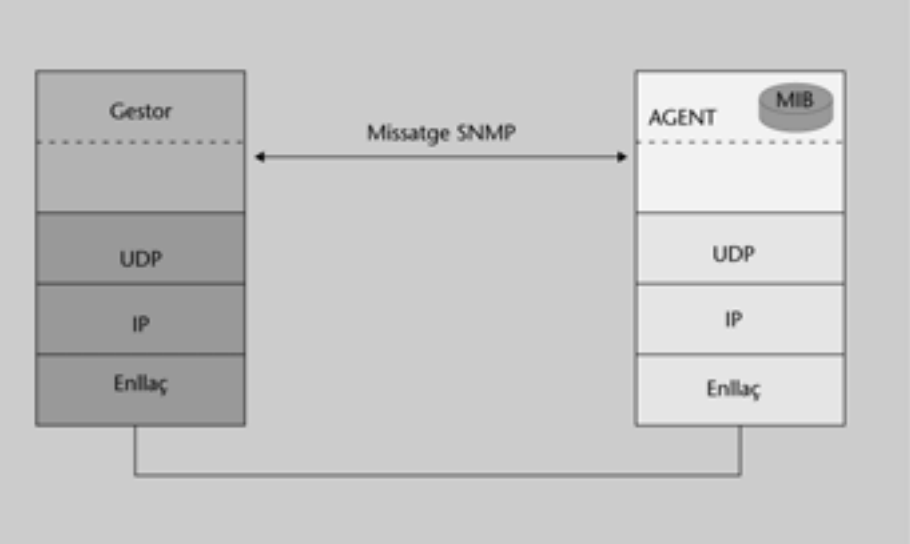
La **base de dades de gestió (en anglès *managament information base*, MIB)** és una base de dades virtual dels recursos per gestionar, els quals es representen com a objectes. Aquests objectes de la MIB estan **estandarditzats**, són accessibles per l'agent i poden ser manipulats via SNMP per realitzar la gestió de xarxa.

En el subapartat 2.3 tractarem en detall la MIB i els seus objectes.

Finalment, l'últim element bàsic dintre de l'arquitectura és el protocol de comunicació, el qual permet comunicar l'estació de gestió amb els agents. En aquest cas és el protocol SNMP. És un protocol que es basa en el paradigma **petició-resposta**.

En la figura 3 tenim l'esquema bàsic amb els quatre elements descrits.

Figura 3. Model d'arquitectura de gestió SNMP



Protocol SNMP

El protocol SNMP està muntat sobre UDP, que és un protocol de capa de transport no orientat a connexió.

2.3. La base de dades d'informació a SNMP

La base de dades d'informació s'anomena MIB (*management information base*) i conté la informació del elements per ser gestionats.

La MIB està estructurada en forma **jeràrquica**, on cada recurs està representat per un objecte.

En el subapartat 2.3.2 podeu veure com està formada l'estructura jeràrquica de la MIB.

La MIB ha de complir un parell de regles bàsiques:

1) L'objecte u objectes utilitzats per a representar un recurs particular ha de ser el mateix per a tots els sistemes.

Seria impossible crear un protocol per adquirir informació si aquesta informació depengués del sistema per gestionar.

2) Ha d'existir un esquema comú de representació per suportar interoperabilitat. Aquest esquema comú és l'**estructura de dades d'informació (SMI).**

2.3.1. Estructura de dades d'informació. Model d'informació

L'SMI defineix el tipus de dades que es poden utilitzar i especifica com es poden representar i anomenar els recursos.

SNMP usa com a notació un subconjunt d'*abstract syntax notation one* (ASN.1) per a acomodar la comunicació entre sistemes heterogenis.

ASN.1 és un format de dades que és comú a totes les màquines i, per tant, que és independent del hardware, sistema operatiu. Per fer aquesta representació de les dades es podia haver agafat algun llenguatge d'alt nivell ja existent, com C, però el que s'ha volgut és que sigui independent del llenguatge.

Implementació de l'ASN.1

L'ASN.1 és una notació comuna per a representar les dades, però la seva implementació es pot fer en el llenguatge que es vulgui.

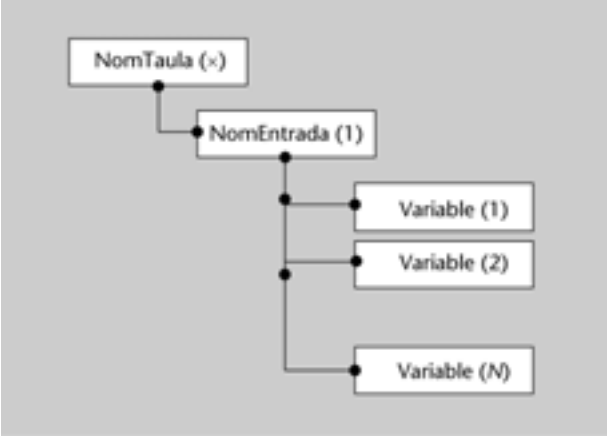
La MIB SNMP emmagatzema, de tots els tipus definits per l'ASN.1, només els tipus de dades simples, entenent per simples el següent:

- **Escalars**. Així les possibles variables són les que es mostren en la taula següent.

Nom	Tipus	Bytes	Significat
INTEGER	Numeric	4	Enter 32 bits
Counter32	Numeric	4	Unsigned 32 bits counter that wraps
Gauge32	Numeric	4	Unsigned value that does no wrap
Integer32	Numeric	4	32 bits
Unteger32	Numeric	4	Like integer32 but unsigned
Counter64	Numeric	8	A 64 bit counter
TimeTicks	Numeric	4	
BIT STRING	String	4	Bit map of 1 to 32 bits
OCTET STRING	String	0	Variable length byte string
Opaque	String	0	Obsolete
OBJECT IDENTIFIER	String	> 0	A list of integers
ipAddress	String	4	A dotted decimal integer address
NsapAddress	String	< 22	An OSI NSAP address

- ***Arrays* bidimensionals d'escalars (taules)**. La manera de definir una taula, que és usada per totes les taules de la MIB, és mitjançant dos tipus ASN.1, que són: SEQUENCE i SEQUENCE OF. SEQUENCE és comparable al que entenem per un *array* en programació, mentre que SEQUENCE OF seria comparable a una estructura. D'aquesta manera, la taula es crea a partir d'un *array* on cada element de l'*array* és una estructura d'escalars. La forma es mostra en la figura 4.

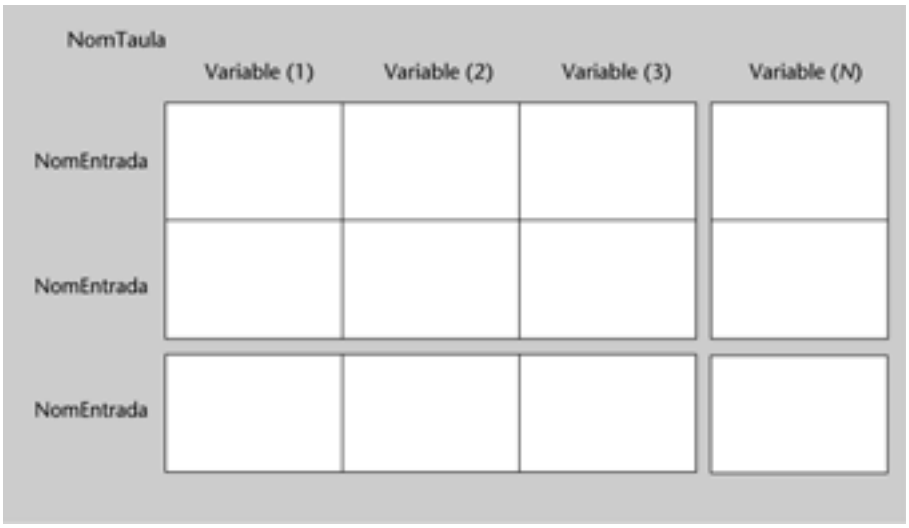
Figura 4. Exemple de taula genèrica SNMP



Les fileres de la taula estaran formades per elements *NomEntrada* els quals són una SEQUENCE. A la vegada hi haurà una de les variables o diverses que seran els índex. Aquests índex permetran distingir cada una de les fileres dintre de la taula (figura 5).

Parlarem més endavant de l'ús dels índexs.

Figura 5. Representació d'una taula SNMP



2.3.2. Estructura de la MIB

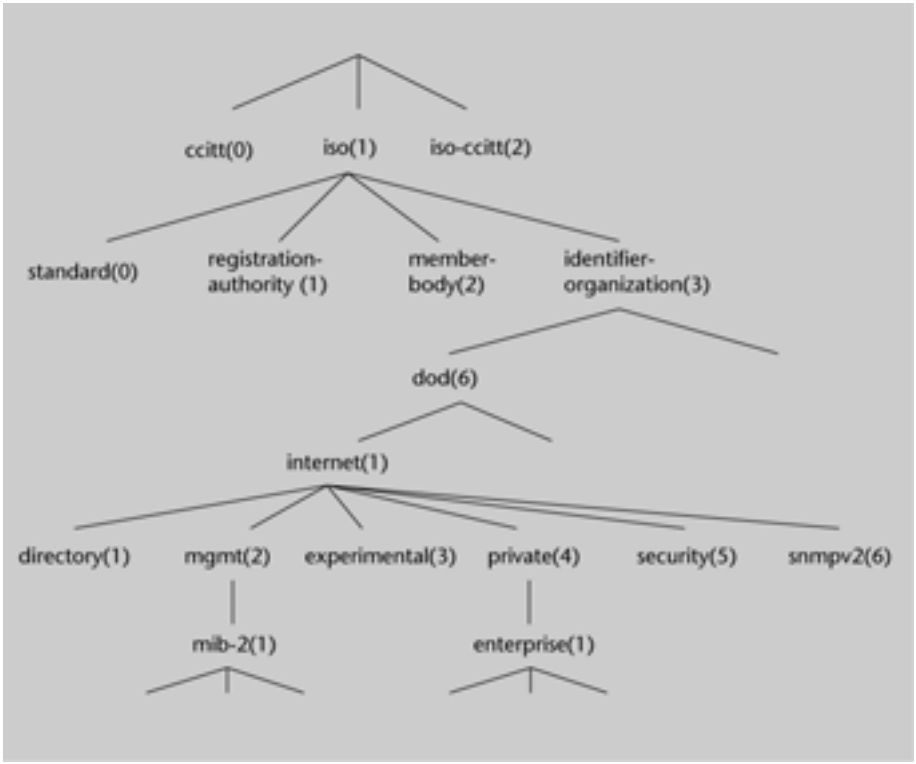
Per a definir l'estructura de la MIB el primer que cal fer és tenir clar com se'n volen definir els objectes. Així l'estructura utilitzada segueix les regles bàsiques següents:

- Identificació no ambigua i universal dels objectes a través d'una arquitectura en forma d'arbre.
- Aquesta estructura està basada en l'esquema d'identificació d'objectes definit per l'OSI.

Seguint aquest esquema de l'OSI els objectes gestionats a l'entorn SNMP estan ordenats en una arquitectura d'arbre on les branques són els diversos objectes.

Associat a cada tipus d'objecte hi ha un identificador de tipus ASN.1 (*object identifier*) que serveix per a anomenar els objectes i per a situar-se dintre de l'estructura de l'arbre. La jerarquia MIB es pot veure com un arbre amb una arrel sense nom i on tenim diversos nivells, que estan assignats per diverses organitzacions (figura 6).

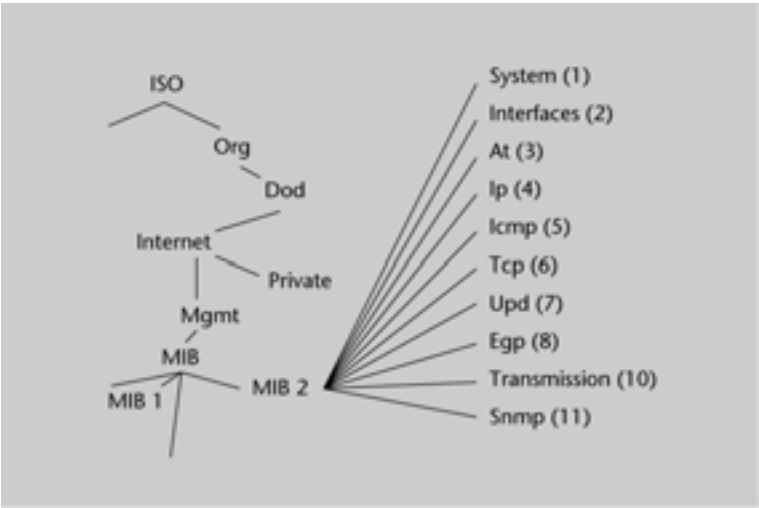
Figura 6. Representació de forma jeràrquica de la MIB



Un aspecte molt important en la comunicació entre dispositius heterogenis és que hi hagi una representació de les dades que sigui comuna. Així aquesta tasca, com ja s'ha dit, la fa la notació estandarditzada ASN.1.

Dintre de la MIB-II trobem els grups que es mostren en la figura 7.

Figura 7. Grups de la MIB 2 dintre de l'estructura jeràrquica



En la taula següent es mostren els deu grups administrats amb una breu descripció de cadascun d'ells.

Grup	Objectes	Descripció
System	7	Nom, localització i descripció de l'equip
Interfaces	23	Interfícies de xarxa i estadístiques de trànsit
At	3	Translació d'adreces (no usat)
Ip	42	Estadístiques dels paquets IP
Icmp	26	Estadístiques de paquets ICMP rebuts
Tcp	19	Algorismes, paràmetres i estadístiques de trànsit TCP
Udp	6	Estadístiques de trànsit UDP
Egp	20	Estadístiques de trànsit EGP
Transmission	0	Reservat
snmp	29	Estadístiques de trànsit SNMP

2.3.3. Definició dels objectes

Una MIB està formada per objectes. Cada objecte definit té associat un tipus i un valor. Així, com podeu imaginar, hi ha molts tipus diferents en funció de la informació dels objectes.

Per senzillesa s'usa una macro per a crear els diferents objectes. Aquesta macro defineix la sintaxi per als diferents conjunts de tipus relacionats.

```
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"
      | "non-accessible"
  Status ::= "mandatory"
      | "optional"
      | "obsolete"
END
```

A partir d'aquesta es creen els tipus específics, el que anomenem "instàncies macro".

```
OBJECT:
  object descriptorobject identifier
SYNTAX:
  sintaxi ASN.1 dels objectes
DEFINITION:
  descripció de l'objecte
ACCESS:
  només lectura, lectura-escriptura o no accessible
STATUS:
  obligatori, opcional o obsolet
```

Exemple de definició d'objecte

A continuació teniu un exemple de definició d'objecte que proporciona l'RFC 1212

```
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 }
```

Web recomanada

Els RFC (*request for comments*) contenen informació tècnica i documents associats a Internet que inclouen especificacions tècniques i documents polítics produïts per l'Internet Engineering Task Force (IETF). Els podeu consultar a http://www.rfc-editor.org.

En el cas de l'exemple, quan es vol anomenar aquest objecte concret cal fer: *Iso (1) identified-organization (3) dod (6) internet (1) mgmt(2) mib-2(1) interfaces (2) IfNumber (1)* o la forma com s'expressa habitualment 1.3.6.1.2.1.2.1.

En el subapartat 2.4.3 podeu veure en detall com es fa la petició a objectes concrets (instàncies).

Per a poder demanar la informació concreta de la variable associada a aquest objecte llavors cal incloure un 0: 1.3.6.1.2.1.2.1.0

Un tipus d'objecte que cal tractar a part són les taules. SMI suporta només una forma d'estructura de dades: **taules bidimensionals d'escalars**.

2.3.4. Les MIB privades

Una de les característiques més importants d'SNMP és el fet que la MIB ha estat dissenyada de manera que pot anar creixent i així proporciona flexibilitat per a incorporar nous objectes. Si us fixeu en la figura 7, on es mostra l'estructura MIB hi ha una branca *private* en què es poden afegir extensions. Això permet als fabricants incorporar objectes associats a la gestió específica dels seus productes. Gràcies a l'estandardització de l'SMI encara que creem objectes privats aquests es poden gestionar des de qualsevol plataforma de gestió, és a dir, hi ha d'haver una interoperabilitat que arribi fins a les extensions privades de la MIB.

Per defecte, les estacions de gestió només coneixen la MIB estàndard. Per tant, per poder gestionar objectes MIB privats cal que prèviament es carregui l'estructura de la MIB privada en la plataforma de gestió. 

Les plataformes de gestió actuals permeten compilar noves MIB per a gestionar recursos privats.

2.4. El protocol SNMP

Hi ha uns conceptes bàsics relacionats amb les operacions del protocol SNMP:

- Les operacions que suporta SNMP
- Les comunitats
- La identificació d'instàncies

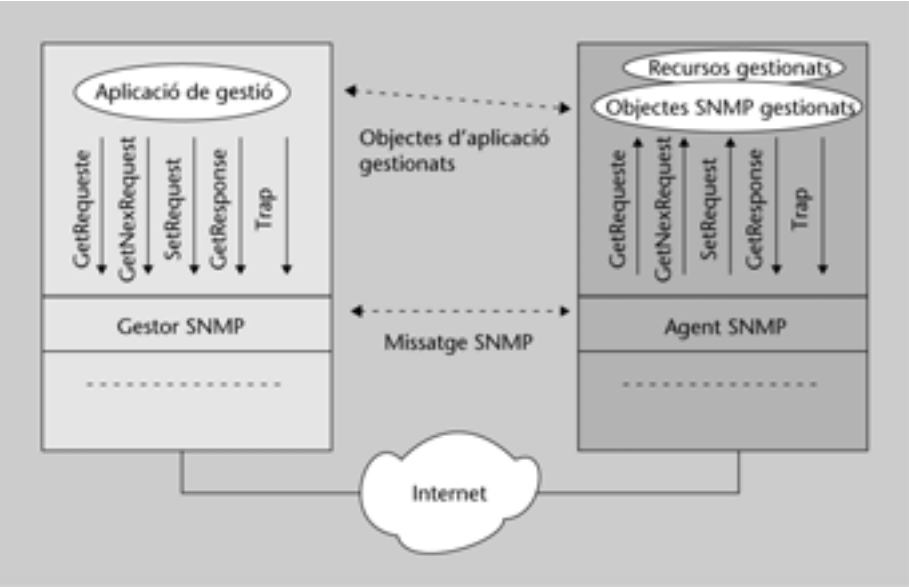
2.4.1. Operacions en SNMP

Totes les operacions que suporta el protocol SNMP estan relacionades amb la modificació i inspecció de variables. En els objectes escalars es poden agrupar en tres tipus d'operacions bàsiques:

- **Get**. L'estació de gestió demana un objecte escalar a l'estació gestionada.
- **Set**. L'estació de gestió actualitza un objecte escalar d'una estació gestionada.
- ***Trap***. L'estació gestionada envia el valor d'un objecte escalar al gestor sense sol·licitud prèvia.

Hi ha cinc tipus de missatges usats per SNMP, com mostra la figura 8.

Figura 8. Accions SNMP



Accés a un únic objecte cada cop

L'accés a un únic objecte cada cop és una restricció que simplifica la implementació d'SNMP però limita la capacitat del sistema de gestió.

No és possible accedir a una taula sencera amb una única acció.

2.4.2. Les comunitats SNMP

La comunitat SNMP és una relació entre agent i gestor que defineix bàsicament **autenticació** i **control d'accés**.

Una comunitat SNMP és una **relació** entre l'**agent SNMP** i els **gestors SNMP**. El concepte de comunitat és **local** i es defineix en el sistema gestionat. Aquest crea una comunitat per a cada combinació d'autenticació i control d'accés. A cada comunitat se li dóna un nom de comunitat únic (dintre de l'agent), el qual es proporciona a l'estació de gestió. Aquesta estació, a la vegada, ha d'usar aquest nom cada cop que fa una operació *get* o *set*. L'agent pot establir diverses comunitats en funció dels permisos dels gestors.

Llavors, **com fa l'autenticació?** Cada missatge que genera el gestor cap a l'agent ha d'incloure el nom de comunitat (*community name*).

Autenticació i control d'accés

Entenent per autenticació el cas en què el missatge és autèntic.

Control d'accés = diferents privilegis en funció del gestor.

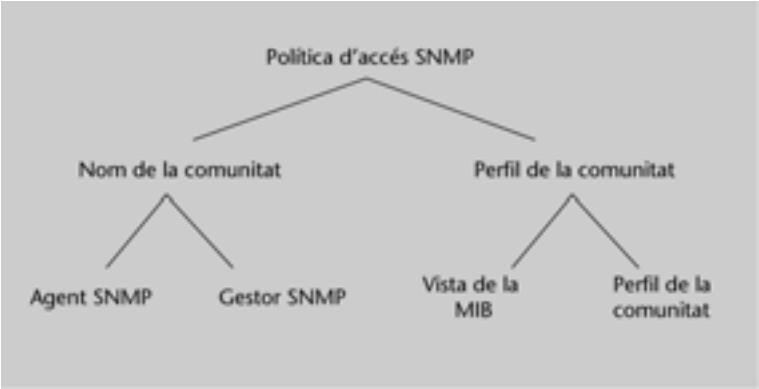
I com funciona la política d'accés? L'agent pot crear diferents categories d'accés a la MIB en funció de qui sigui el gestor. El que fa és crear diferents perfils lligats al nom de comunitat que tenen accés a les diferents parts de la MIB amb permisos concrets de lectura o lectura/escriptura (mode d'accés).

Així, la figura 9 intenta mostrar tot el que serien les polítiques d'accés a SNMP. Per una banda, tenim el nom de comunitat que coneixen tant l'agent com el gestor. Per l'altra, el que tenim és que cada comunitat té associat un perfil que li permet tenir accés a unes parts de la MIB concretes.

Noms de comunitat dels dispositius

Els dispositius per defecte solen portar dos noms de comunitat coneguts per tothom: *public* (lectura) i *private* (lectura/escriptura). Cal tenir cura d'eliminar-los.

Figura 9. Conceptes administratius



2.4.3. Identificació d'instàncies

Cal que recordeu la diferència entre objecte i instància de l'objecte.

Cada objecte a la MIB té un únic identificador d'objecte, que està definit per la posició que ocupa dintre de l'arbre MIB. No obstant això, quan es fa un accés a la MIB via SNMP el que es vol no és el tipus de l'objecte sinó una instància concreta (el valor de la variable). Vegem com es fa.

Cas d'objectes escalars

Per a obtenir la instància d'un objecte escalar cal posar el camí per a arribar a l'objecte i incloure al final sempre **.0.**

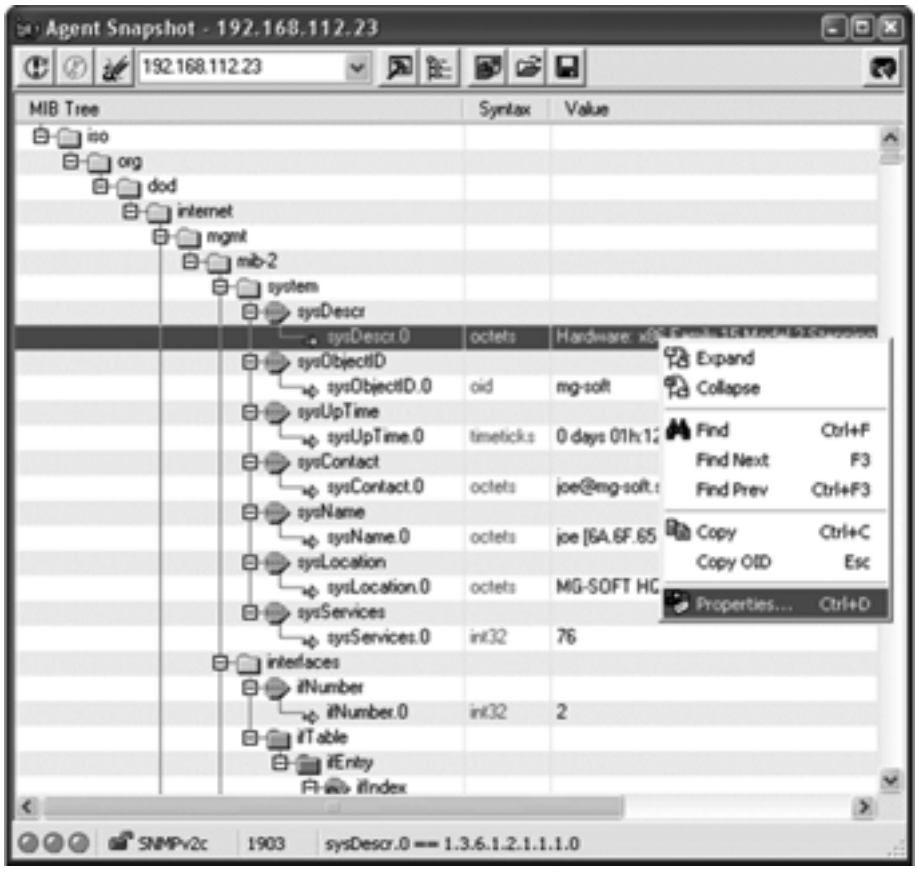
El 0 es posa per diferenciar entre fer una referència a un objecte i fer-la a la variable associada a aquest objecte.

Per exemple tenim,

	OBJECT IDENTIFER
Snmp subtree	1.3.6.1.2.1.11
snmpInBadValues	1.3.6.1.2.1.11.10
instància de snmpInBadValues	1.3.6.1.2.1.11.10.0

En la figura 10 es mostra l'arbre mib-2 i unes peticions als diversos objectes del grup *system*.

Figura 10. Resultat obtingut del grup *system* amb una aplicació de gestió



Objectes de tipus taula

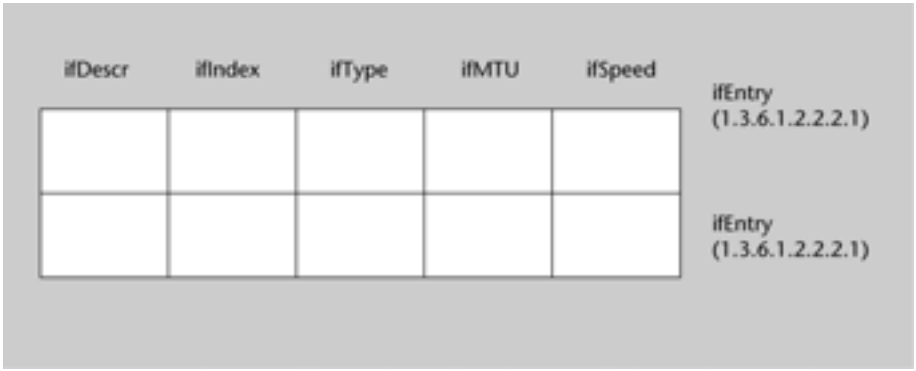
L'accés a les instàncies dintre d'una taula, encara que el mecanisme és senzill, resulta una mica complicat d'explicar. El millor és veure-ho mitjançant un exemple.

Exemple

Suposem un dispositiu amb dos interfícies i volem informació del grup dintre de la mib-2 anomenada *interface*. Aquest grup té la forma següent:

```
Interfaces
  ifTable
         ifEntry
                ifDescr
                IfIndex
                ifType
                ifMTU
                ifSpeed
```

En aquest cas tindríem una taula amb dos files i cinc columnes de la forma següent:



A partir d'aquesta informació, si el que volem fer és una instància concreta, per exemple d'*ifType*, haurem de fer:

En cas que tinguem una taula amb més d'un índex caldria fer:

Ruta.objecte.valor_index1.valor_index2_..valor_index*n*

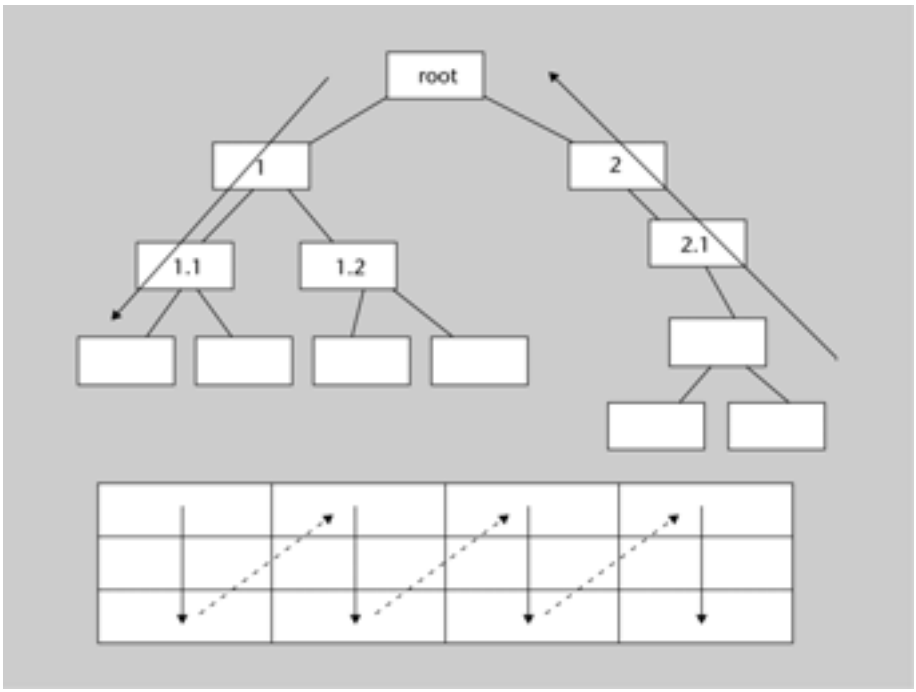
Aquesta és l'única forma d'identificar correctament la filera exacta on es troba la instància de l'objecte sol·licitat.

Cal que tingueu present que els índex poden tenir qualsevol valor en funció de l'objecte que representin.

Es molt interessant tenir clar l'ordre lexicogràfic en un arbre d'objectes, tant escalars com en una taula.

En la figura 11 es mostra com ens movem dintre de l'arbre jeràrquic i a la vegada dintre d'una taula concreta si anem sol·licitant els valors dels objectes un darrere l'altre.

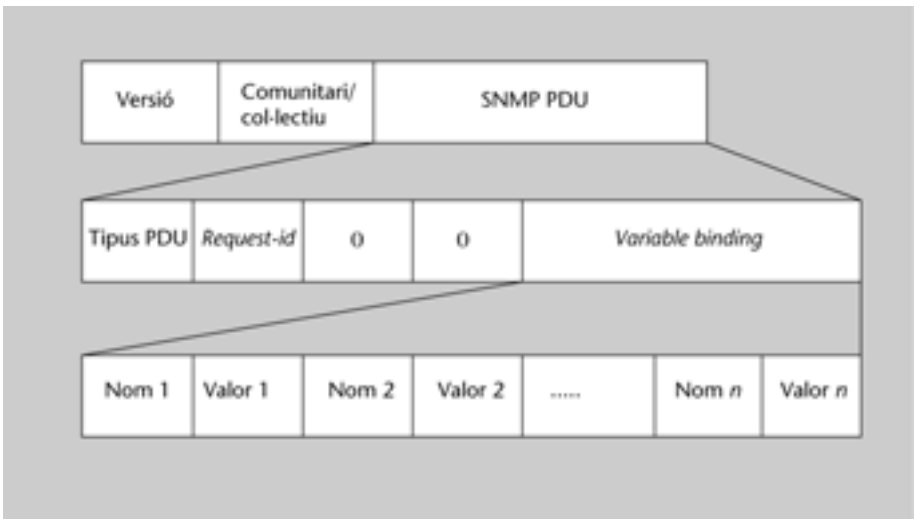
Figura 11. Ordre d'accés a les instàncies dels objectes dintre de les taules SNMP



2.4.4. Format del paquet SNMP

En la figura 12 es mostra el format del paquet SNMP. Com podem veure tots els missatges inclouen el número de versió d'SNMP, nom de la comunitat per utilitzar en l'intercanvi i tipus de missatge.

Figura 12. Format dels paquets SNMP



Cal saber que en un mateix missatge SNMP es poden incloure diverses operacions del mateix tipus.

Un cas interessant és *GetNextRequest*. Es similar a *GetRequest* però amb la diferència que permet veure el valor següent dintre de l'estructura de l'arbre, és a dir, quan sol·licitem una instància no obtenim el valor de la mateixa instància

sinó el valor corresponent a l'objecte immediatament posterior al sol·licitat. Es molt útil en el cas que es vulgui accedir als valors d'una taula. Si recordeu, hem explicat que per a accedir a les instàncies d'una taula calia posar els valors de les instàncies índex d'aquesta taula, la qual cosa és complicada *a priori*, ja que no sabem aquests valors. En aquest cas, el que fem és demanar el valor justament anterior dintre de l'arbre d'objectes i el que l'agent ens retorna és el primer valor de la taula.

Exemple

Si recordem l'exemple anterior a la petició següent: GetNext (1.3.6.1.2.2.1.0), la resposta que obtindrem serà:

GetReponse = 1.3.6.1.2.2.2.eth0.1

En aquest cas ens retorna el valor d'*IfDescr* i a la vegada el valor que té l'índex de la primera fila.

2.5. Limitacions del protocol SNMP

Un cop vist el funcionament bàsic del protocol SNMP, cal que tingueu present algunes de les seves limitacions:

- SNMP no és adequat per a la gestió de xarxes molt grans ateses les limitacions de rendiment lligades al paradigma petició-resposta.
- SNMP no està ben dissenyat per a grans volums de dades com són les taules.
- Els *traps* no tenen reconeixement, la qual cosa pot provocar que *traps* crítics es puguin perdre sense que arribin al gestor. Recordeu que els *traps* permeten a l'agent generar un missatge cap al gestor sense petició prèvia.
- L'autenticació, com s'ha vist, és molt trivial.
- La MIB té limitacions.
- SNMP no suporta comunicació entre gestors.

Per què els *traps* no tenen reconeixement

Doncs, com s'ha dit, perquè SNMP té UDP com a protocol de capa de transport.

3. SNMPv2

L'SNMPv2 és una evolució de la versió inicial d'SNMP, la qual inclou tota una sèrie de millores respecte a la versió anterior.

3.1. Millores en la gestió

Cal que tingueu present que el primer que cal destacar és el canvi en la filosofia de treball d'aquesta nova versió. SNMPv2 pot suportar tant **gestió centralitzada** com **distribuïda**. En aquest cas alguns gestors faran el paper de gestor i agent simultàniament. Com a agents acceptaran comandes d'un gestor de categoria superior. Aquestes peticions poden ser per a obtenir informació del gestor intermedi o per a demanar informació en aquest gestor un informe dels agents que hi estan subordinats. De la mateixa manera, el gestor intermedi pot enviar *traps* al superior.

Les millores d'SNMPv2 s'agrupen en tres categories:

- Estructura de la informació de gestió
- Capacitat gestor-gestor
- Operacions del protocol

3.2. Estructura de la informació de gestió

L'estructura de la informació de gestió (SMI) creix respecte a l'SMI de la versió 1. Trobem una estructura amb més tipus de dades, millora en la documentació associada a cada objecte i un aspecte molt important que és una nova convenció per a crear i esborrar fileres en les taules de la MIB.

Si hem de resumir breument com queda aquesta estructura es pot reduir a quatre conceptes clau:

1) Definició d'objectes. Es crea una nova macro que permet incloure nous tipus de dades i millorar la documentació associada a cada objecte.

2) Taules conceptuals. SNMPv2 millora el mecanisme usat tant a l'RFC 1212 com a l'especificació RMON per a facilitar la creació, eliminació i accés a les fileres de les taules.

3) Definicions de notificació. Es crea una macro específica que defineix la informació enviada per una entitat SNMPv2 quan hi ha un esdeveniment excepcional en l'entitat.

4) Mòduls d'informació. SNMPv2 introdueix el concepte de mòdul d'informació, que especifica un grup de definicions que estan relacionades.

3.3. El protocol SNMPv2

Quant a les operacions del protocol, aquest inclou dos nous tipus de PDU:

- *GetBulkRequest*. Permet obtenir un gran nombre de blocs de dades de forma eficient, com són la possibilitat d'obtenir diverses fileres d'una taula.

- *InformRequest*. Amb aquesta PDU el gestor intermedi pot enviar *traps* a un altre gestor.

4. RMON (*remote network monitoring*)

L'objectiu d'RMON és proporcionar informació a la plataforma de gestió dels diferents segments de xarxa però sense canviar el protocol SNMP. Per a fer-ho, podem considerar que dintre de la xarxa tindrem una sèrie de dispositius que seran capaços de monitoritzar la informació, analitzar-la i comunicar-se de forma adequada amb el gestor. Això, d'entrada, evita la necessitat de conèixer el trànsit d'entrada i sortida de cadascun dels dispositius, la qual cosa en la majoria de casos no és interessant.

Tradicionalment els equips que han analitzat el trànsit han estat els analitzadors de xarxa o en ocasions anomenats detectors (*sniffers*).

Actualment molts dels commutadors (*switchs*) de gamma mitjana i alta incorporen prestacions RMON.

4.1. Objectius del disseny d'RMON

Els objectius del disseny d'RMON estan especificats en l'RFC 1757 i són bàsicament els següents:

- **Limitar el sondeig (*polling*) permetent fer operacions fora de línia**. Aconseguim millorar el rendiment, continuar recollint informació del segment en cas que caigui la línia entre gestor i monitor, i quan aquesta es recupera enviar la informació.

- **Monitoratge proactiu**. Si el monitor té prou recursos és interessant que pugui treure diagnosis i informes de rendiment de la xarxa.

- **Detecció de problemes i tramesa d'informes (*reports*)**.

- **Possibilitat de diàleg amb més d'un mànager**. Això millora la fiabilitat.

- **Anàlisi del trànsit recollit.**

4.2. Control de la sonda RMON

El dispositiu que suporta RMON s'anomena normalment sonda RMON o simplement **sonda**.

La sonda RMON té alguna característica pròpia a l'hora de ser gestionada, ja que és capaç de fer feines més complexes que un agent amb MIB-2. Aquestes característiques estarien associades a la **configuració de la sonda** i a la **invocació d'accions**.

4.2.1. Configuració de la sonda

Una sonda necessita ser configurada per a recollir dades. En funció de la configuració que fem ens permetrà decidir el tipus i la forma de les dades recollides. La forma de fer-ho s'explica a continuació.

La MIB RMON està estructurada en grups com a MIB-2, però amb la particularitat que tots els grups estan formats per taules. Així per a cada grup hi ha **taules de control** i **taules de dades**. Les taules de control són taules de lectura/escriptura i contenen paràmetres que descriuen les dades. Les taules de dades són només de lectura. Si ho penseu té sentit que aquestes taules només siguin de lectura. El que fa l'estació de gestió és configurar els paràmetres de control apropiats perquè la sonda reculli les dades que interessen al gestor en cada moment.

Sembla lògic que la MIB tingui només taules, ja que el que fa és obtenir informació dels segments de xarxa.

Els paràmetres es posen incloent una nova fila o modificant una fila de la taula de control. Per tant, les funcions que ha d'efectuar una sonda estan definides i implementades en termes de files d'una taula.

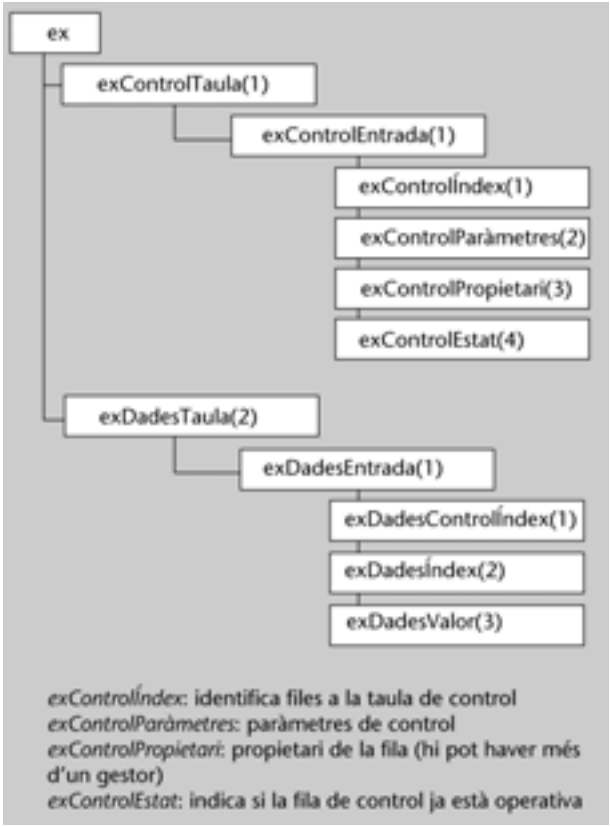
Per exemple, en la taula de control es poden indicar paràmetres com origen de les dades, tipus, temps entre captures, etc. Així l'índex en la taula de control serveix per a accedir a la taula de dades, i a l'inrevés.

La millor manera d'entendre el funcionament és mitjançant un exemple.

Exemple de configuració d'una sonda

Suposem el grup fictici *ex* (exemple) amb l'estructura que es mostra en la figura 13.

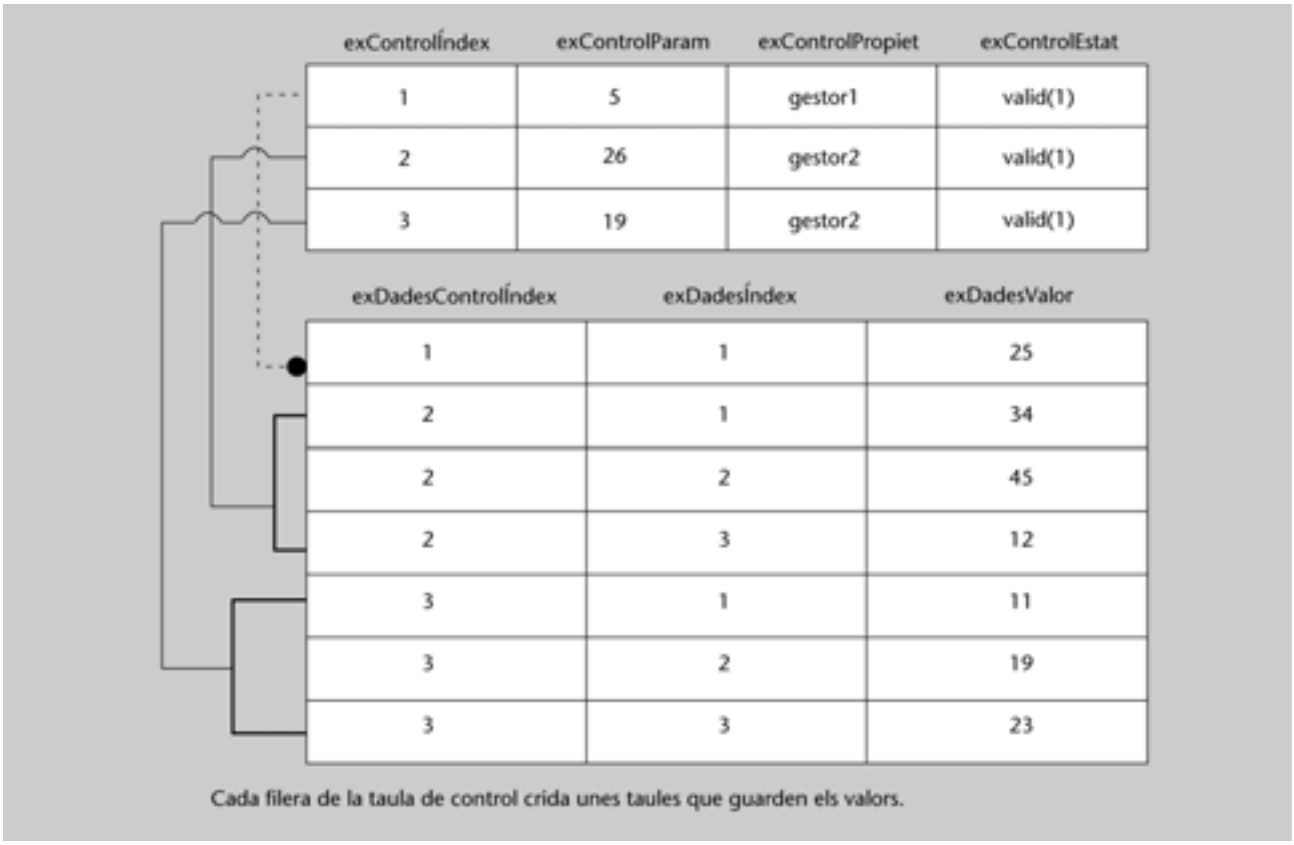
Figura 13. Representació d'un grup genèric seguint el format RMON



La taula de dades té dos índex, on *exDadesControlIndex* coincideix amb *exControlIndex*.

Així, una possible configuració de les taules es mostra en la figura 14.

Figura 14. Valors concrets d'una possible taula resultant

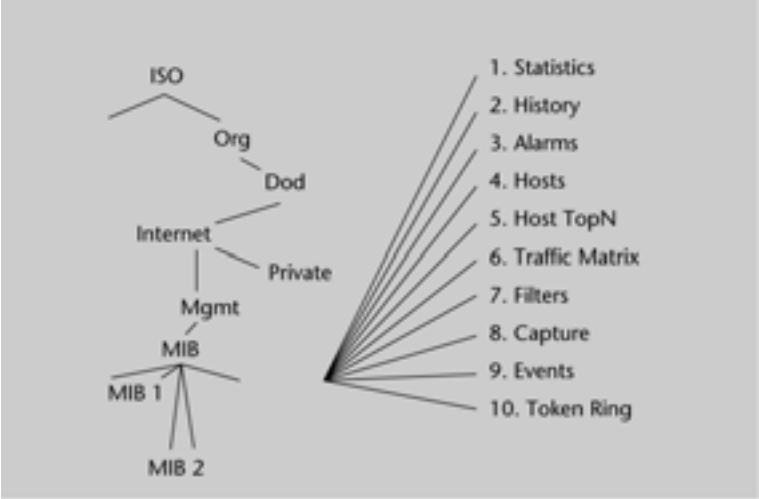


L'índex comú de les dues taules permet identificar les files de dades associades a un gestor concret creat en la taula de control.

4.3. RMON1 i RMON2

Inicialment RMON va ser desenvolupat per a xarxes LAN Ethernet i Token Ring de manera que divideix les funcions de monitoratge en nou grups associats a topologies Ethernet i un grup deu per a Token Ring. En la figura 15, es veuen els deu grups que formen part de la branca MIB.

Figura 15. Grups RMON1 dintre de l'estructura MIB



L'estàndard RMON està acceptat extensament per les empreses de *networking* i comunicacions.

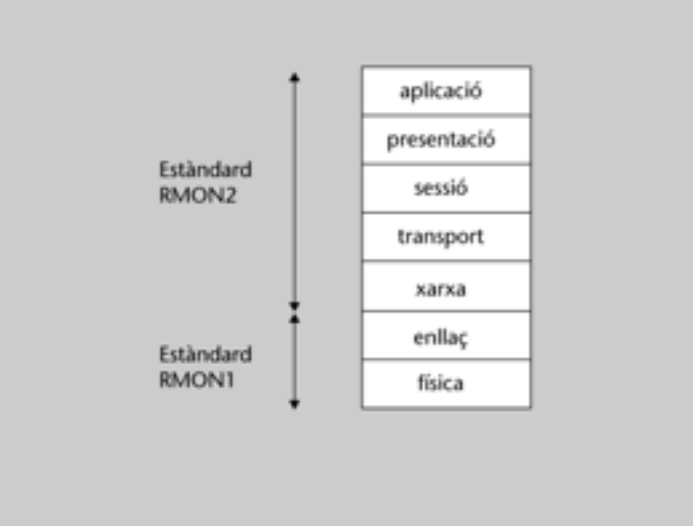
Una limitació important d'RMON1 és que especifica només monitoratge i diagnòstic de trànsit de xarxa a nivell de capa d'enllaç i no fa el monitoratge de trànsit a nivell d'aplicacions extrem a extrem.

Com a conseqüència d'aquesta limitació ens trobem que RMON1 no pot identificar dispositius que estiguin fora de la seva subxarxa, és a dir, darrere de l'encaminador.

Penseu atentament en el que això implica.

RMON2 respon a la necessitat d'analitzar el trànsit de les capes superiors. Es pot considerar una extensió d'RMON1 i són tecnologies complementàries com es pot veure en la figura 16.

Figura 16. Capes OSI on aplica cadascun dels protocols RMON1 i RMON2



En la taula següent podem veure com es complementen ambdós estàndards per a cobrir els diferents temes de monitoratge de xarxa.

Aspectes de la gestió	Nivell OSI	Estàndard RMON
Utilització i errors físics	Control d'accés al medi	RMON1
Segmentació LAN	Enllaç	RMON1
Interconnexió de xarxes	*Network*	RMON2
Ús d'aplicació	Aplicació	RMON2

4.4. La MIB RMON

Com s'ha dit, la MIB RMON està dividida en deu grups. En la taula següent es mostra una descripció de cadascun dels grups.

Grup	Descripció
Statistics	Manté estadístiques d'utilització i errors dels nivells baixos per a cada subxarxa monitoritzada.
History	Guarda mostres d'estadístiques de forma periòdica. Les extreu del grup *statistics*.
Alarm	Permet al gestor configurar intervals de mostratge i llindars d'alarma associats als valors obtinguts per la sonda.
Host	Conté comptadors de diversos tipus de trànsit dels *hosts* de la subxarxa.
hostTopN	Conté estadístiques dels *hosts* ordenades.
Matrix	Ensenya informació d'errors i utilització en forma de matriu.
Filter	Permet veure certs paquets a partir d'un filtre concret.
Capture	Controla com s'envien les dades a la consola de gestió.
Event	Dóna una taula amb els esdeveniments generats en la sonda.
TokenRing	Manté estadístiques i informació de configuració per a les subxarxes Token Ring.

A la vegada, els grups RMON els podem agrupar en funció de la tasca que fan. Podem trobar dos grups de tasques bàsiques:

- Estadístiques de trànsit
- Alarmes, filtres i captura de paquets

Alarmes, filtres i captura de paquets

Encara que dintre del grup d'alarmes, filtres i captura de paquets hi ha tres tasques, veurem que depenen unes de les altres pel seu correcte funcionament.

Tots els grups són opcionals en la seva implementació, però hi ha dependències entre alguns d'ells. Així, les dependències principals són les següents:

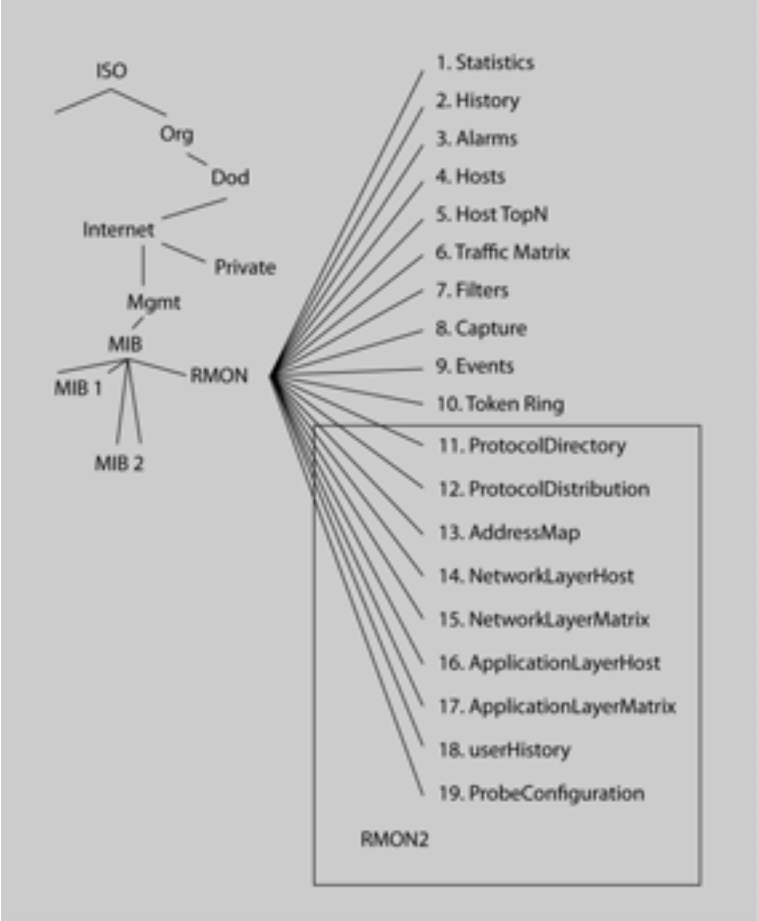
- El grup *alarm* requereix la implementació del grup *event*.
- El grup *hostTopN* requereix la implementació del grup *host*.
- El grup *capture* requereix la implementació del grup *filter*.

4.4.1. RMON2

Com hem dit anteriorment **RMON2** descodifica paquets de **nivell 3 a 7** de la pila OSI. Això vol dir que pot fer **monitoratge de trànsit a nivell IP** i, per tant, no està limitat a un segment Ethernet com passava en RMON1. Podem saber cap on va el trànsit que va cap a fora de la pròpia subxarxa i el mateix succeeix amb el trànsit entrant. Fins ara aquest trànsit a nivell 2 sempre tenia l'adreça MAC de l'encaminador per defecte. El mateix passarà amb la descodificació a nivell d'aplicació. Podrem saber cap a quina aplicació es genera el trànsit de cadascuna de les subxarxes.

Tal com es pot veure en la figura 17, la MIB RMON2 és una extensió de la MIB RMON.

Figura 17. Grups RMON2 dintre de l'estructura MIB



Altres característiques d'RMON2

Les noves característiques d'RMON2 estan associades a la forma d'**indexar de les taules**. Un gestor el que fa és sol·licitar la informació de les sondes de forma periòdica amb l'inconvenient quant a eficiència que implica el fet que cada cop retorna els nous valors a més dels antics. El que incorpora RMON2 és un índex temporal que permet enviar només el valors dels objectes que han canviat des de l'última petició a la sonda RMON.

RMON2 no envia tantes dades i per tant fa un ús més eficient de l'amplada de banda.

5. Gestió TMN

Les operadores de telecomunicacions requereixen una infraestructura de gestió singular. Aquest entorn de gestió ha de preveure com a objectiu últim la prestació de servei de manera flexible i dinàmica. El model clàssic de gestió orienta les seves tasques cap a la xarxa, és a dir, configuració, manteniment i si és possible anàlisi de rendiment. La necessitat de les capacitats de gestió cap a la banda dels clients i els serveis ha provocat l'aparició de models de referència per a desenvolupar la gestió TMN.

Fa uns anys la indústria de les "telecos" usava solucions propietàries per a la gestió. Les operadores donaven nous serveis en funció de les seves possibilitats i per tant anaven millorant el seu sistema de gestió Actualment, amb la desregularització del sector l'increment de la competència i nous serveis les operadores s'han trobat que la gestió propietària no proporciona interoperabilitat entre les diverses tecnologies i les solucions de gestió que incorporen. Un altre aspecte que han de tenir en compte és la coexistència de les noves solucions amb sistemes "antics".

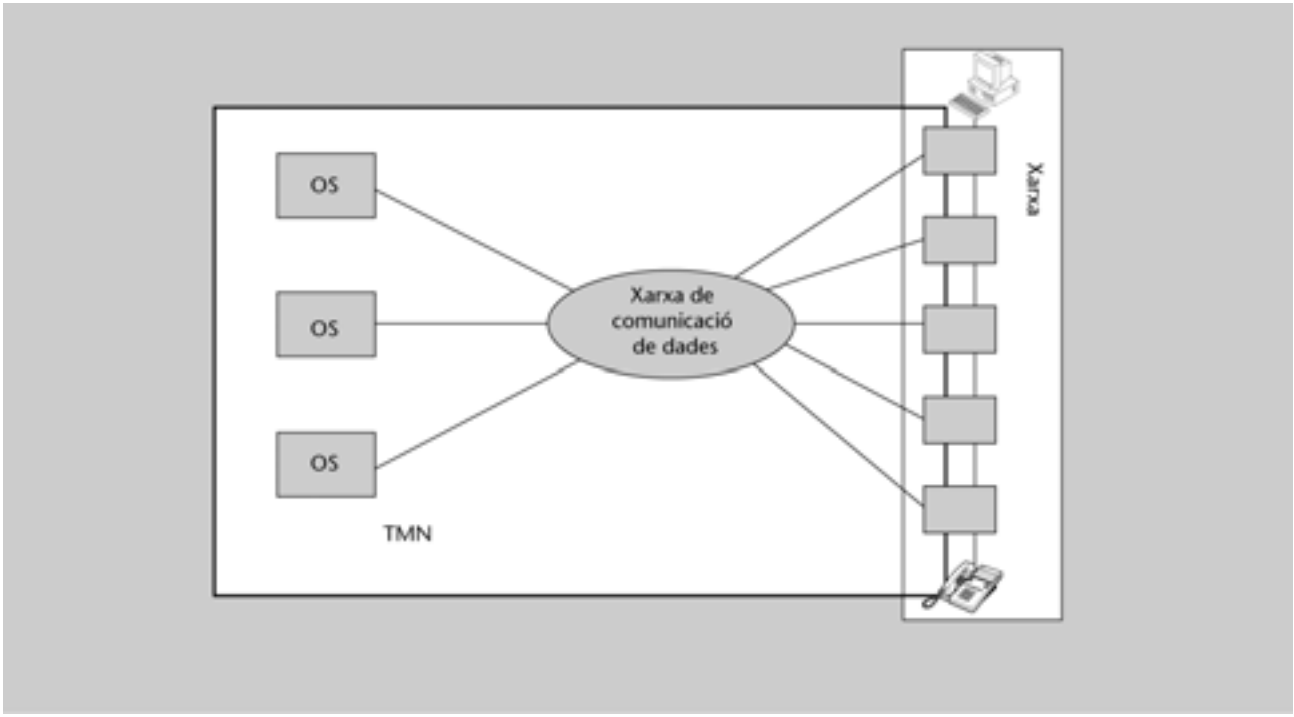
Així, com a resum podem dir que la motivació de l'arquitectura TMN ha estat, d'una banda, l'heterogeneïtat de les xarxes de telecomunicacions i, de l'altra, les demandes sobre aspectes com els següents:

- Possibilitat d'introduir nous serveis
- Alta qualitat dels serveis
- Possibilitat de reorganitzar les xarxes
- Mètodes eficients de treball per a operar les xarxes
- Competència entre operadores

Els principis del TMN (*telecomunications management networks*) es tracten en la recomanació M.3010 de la ITU-T, que **defineix un model d'operació per capes**: capes d'operació. Té una forta relació amb el model de gestió OSI.

L'**objectiu de TMN** és proporcionar una estructura de xarxa organitzada per aconseguir la interconnexió dels diversos tipus de sistemes d'operació i equips de telecomunicació usant una arquitectura estàndard i interfícies normalitzades.

Figura 18. Relació de TMN en una xarxa de telecomunicacions



A diferència del model OSI que definia cinc àrees funcionals, l'estàndard TMN no entra en consideracions sobre aplicacions de la informació gestionada. Per contra, es defineixen les funcions següents:

Podeu consultar les cinc àrees funcionals del model OSI en el subapartat 1.1.1 d'aquest mòdul.

- L'intercanvi d'informació entre la xarxa gestionada i la xarxa TMN
- L'intercanvi d'informació entre xarxes TMN
- La conversió de formats d'informació per un intercanvi consistent de la informació
- La transferència d'informació entre punts d'una TMN
- L'anàlisi de la informació de gestió i la capacitat d'actuar en funció seva
- La manipulació i presentació de la informació de gestió en un format útil per a l'usuari de la mateixa informació
- El control d'accés a la informació de gestió per part dels usuaris autoritzats

No oblideu que el significat de gestió de xarxa per TMN és més ampli que el que havíem considerat en el cas d'SNMP. En ambdós casos, gestió de xarxa significa gestió de xarxes i serveis, però en TMN està focalitzat en xarxes de telecomunicacions, equips i serveis proporcionats als clients.

Protocols que cobreix TMN

Alguns exemples de protocols o tecnologies que cobreix TMN serien: ATM, SDH, SONET o XDSL.

En la indústria de les telecomunicacions, els serveis i configuracions dels usuaris finals (clients) estan inclosos com a part de TMN. Així, apareixen dos termes nous, com són **servei** i **aprovisionament de recursos**.

TMN proporciona el suport de gestió per a la planificació, aprovisionament, instal·lació, manteniment, operació i administració de xarxes i serveis de telecomunicacions.

Una xarxa de telecomunicacions pot incloure una gran varietat de components, com són equips per a transmissió analògica, digital o sense fils (*wireless*), etc.

5.1. Arquitectura TMN

L'arquitectura TMN està dividida en tres blocs. La recomanació M.3010 defineix els blocs següents:

- **Arquitectura física**. Estructura i entitats de la xarxa: com s'implanten les funcions de gestió en els equips físics.
- **Arquitectura funcional**. Components i funcions de gestió.
- **Arquitectura de la informació**. Nivells de gestió: la gestió s'estructura segons responsabilitats.

Adreça recomanada

Podeu trobar un interessant *tutorial* sobre l'arquitectura TMN a http://www.simpleweb.org/tutorials/tmn/

5.1.1. Arquitectura funcional del TMN

L'arquitectura funcional TMN divideix el domini TMN en diferents blocs funcionals. Cada bloc funcional executa una funció específica de gestió. Els blocs són els següents:

- ***Operations system function*** **(OSF)**. Proporciona les funcions de planificació i gestió per a la xarxa de telecomunicacions i per als mateixos components.
- ***Network element function*** **(NEF)**. Monitoritzat i controlat per TMN.
- ***Workstation function*** **(WSF)**. Permet que l'usuari pugui veure la informació.
- ***Mediation function*** **(MF)**. Una mena de passarel·la per a l'intercanvi d'informació de gestió quan els blocs funcionals tenen diferents punts de referència.
- ***Q adaptor function*** **(QAF)**. Per a traslladar informació de gestió entre punts de referència TMN i no TMN.

Un element important que ha sortit referenciat i que encara no s'ha comentat són els punts de referència (*reference points*). Aquests punt són els límits conceptuals dels diferents blocs.

5.1.2. Arquitectura física del TMN

L'arquitectura física explica la implementació dels blocs funcionals en sistemes físics i la interfície entre ells. Els components de l'arquitectura física són:

- ***Operations sytem*** **(OS)**. És l'equivalent al gestor. Dóna suport per al processament d'informació relacionat amb operacions, administració, manteniment i aprovisionament de les xarxes de telecomunicacions.

- ***Data communications network*** **(DCN)**. Capacitat d'encaminament i transport per a l'intercanvi d'informació entre OS i OS, OS i NE, WS i OS, i WS i NE.

- ***Mediation device*** **(MD)**. Té funcions de retransmissió o passarel·la.

- ***Workstation*** **(WS)**. Punt d'entrada o sortida que permet als operadors del sistema accedir a les dades de gestió.

- ***Network element*** **(NE)**. El que coneixem per agent.

- ***Q adaptor*** **(QA)**. Converteix les dades no TMN en dades TMN, i viceversa.

5.1.3. Arquitectura d'informació del TMN

L'arquitectura d'informació explica com els sistemes de gestió OSI i els principis X.500 poden ser aplicats a TMN. L'arquitectura d'informació descriu els recursos que han de ser gestionats per TMN usant les guies de referència per a la definició dels objectes gestionats i la sintaxi ASN.1

Resum

En aquest mòdul didàctic s'ha intentat donar una breu visió del conceptes bàsics de la gestió de xarxa i el perquè de la seva necessitat en un entorn com l'actual. Per fer-ho, s'han explicat els conceptes generals de la gestió i posteriorment s'ha aprofundit en els protocols més usats.

S'ha intentat separar el que és la gestió de xarxa en el que hem anomenat xarxes Internet del que són les xarxes de les operadores o xarxes de telecomunicacions.

Com s'ha vist, el protocol dominant en el món Internet és el protocol SNMP, el qual ha anat evolucionant per cobrir les diferents necessitats que han anat sortint.

Inicialment SNMPv1 era un protocol senzill que treballava de forma centralitzada amb el paradigma petició-resposta. Posteriorment, de la mateixa manera que les xarxes es fan més complexes, el protocol ha hagut d'evolucionar per solucionar altres aspectes no plantejats inicialment. Així SNMPv2 inclou mecanismes que permeten optimitzar la informació emmagatzemada en les taules dels diversos agents. Finalment, encara que no s'ha comentat, hi ha una nova versió SNMPv3 que el que incorpora són mecanismes de seguretat.

Paral·lelament dintre de la MIB s'ha incorporat RMON, que com ja heu vist permet monitoritzar les xarxes sense haver de recorre a sol·licitar informació de cadascun dels dispositius via SNMP.

Finalment, per acabar el mòdul s'han vist les necessitats de les operadores de telecomunicacions i les possibilitats que els ofereix l'estàndard TMN per a cobrir les seves necessitats de gestió, que en la majoria de casos són bastant diferents de les necessitats de les xarxes convencionals.

Activitats

1. Descarregueu d'Internet l'RFC 1212 i l'RFC 1213 i familiaritzeu-vos amb la notació ASN.1 utilitzada per a definir les diferents MIB.

2. Realitzeu un *ping* en un dispositiu de la xarxa on us trobeu i intenteu determinar els valors obtinguts.

3. Amb l'ajuda d'MG SOFT MIB Browser intenteu determinar els diferents objectes de la MIB-2. Podeu trobar una versió d'avaluació d'aquest programa a http://www.mg-soft.com

4. Penseu el motiu pel qual hi ha taules SNMP que necessiten més d'un índex.

5. Esbrineu quines són les plataformes de gestió de dos fabricants com poden ser HP i Cisco.

6. Raoneu per quin motiu es va optar per treballar amb UDP i no TCP en gestió SNMP.

Exercicis d'autoavaluació

1. Tenim les peticions del grup *rs232* següents:

GetNext rs232.rs232Number.0
Response: rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.1.1 = 1

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.1.1
Response rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.2.2 = 2

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.2.2
Response: rs232.rs232InSigTable.rs232InSigEntry.rs232InSigState.1.1 = 2

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigState.1.1
Response rs232.rs232InSigTable.rs232InSigEntry.rs232InSigState.2.2 = 2

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigState.2.2
Reponse rs232.rs232InSigTable.rs232InSigEntry.rs232InSigChanges.1.1 = 3

Getnext rs232.rs232InSigTable.rs232InSigEntry.rs232InSigChanges.1.1
Response Reponse rs232.rs232InSigTable.rs232InSigEntry.rs232InSigChanges.2.2 = 4

A partir de la informació donada per aquestes peticions us demanem el següent:

a) Forma de la taula (contingut de la taula).
b) Quins són els objectes índex?
c) Si fem un *GetNext* a partir de l'últim valor obtingut, quin objecte obtindrem?
d) Construïu l'arbre de l'objecte definit.

Nota: El nom del segon índex és rs232InSigIndex.

2. Cerqueu l'RFC 1213 on està definida la MIB-2 i a partir d'aquesta determineu l'arbre del grup UDP.

Solucionari

1.

a) En aquest exercici el que veiem és que es fa un *Getnext*. Es fa perquè volem accedir a una taula SNMP. Com s'ha explicat, per fer la petició cal saber el valor dels índex corresponents. Com això no és així, amb el *Getnext* la resposta ens dóna el primer valor de la taula i de forma indirecta els valors (contingut) dels índexs de la primera filera de la taula.

La primera resposta

Response: rs232.rs232InSigTable.rs232InSigEntry.rs232InSigPortIndex.1.1 = 1

Ens diu:

Primer valor de la taula *rs232InSifPortIndex = 1* i a més ens diu que la taula té dos índex amb valors 1 i 1.

A partir d'aquí els successius *Getnext* van recorrent tota la taula de la mateixa manera que es mostra en la figura 11. El resultat és el següent:

InSIgPortIndex	InSigState	InSigChanges	InSigIndex
1	2	3	1
2	2	4	2

b) Els objectes índex són *InSIgPortIndex* i *InSigIndex*.

c) S'obtindria el valor *d'InSigIndex* de la primera filera de la taula.

d)
```
rs232
    rs232InSigTable
        rs232InSigEntry
            rs232InSigPortIndex (1)
            rs232InSigState (2)
            rs232InSigChanges (3)
            rs232InSigIndex (4)
```

2. El grup UDP en l'RFC 1213 té la forma següent:

```
udpInDatagrams OBJECT-TYPE
    SYNTAX  Counter
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
            "The total number of UDP datagrams delivered to
            UDP users."
    ::= { udp 1 }

udpNoPorts OBJECT-TYPE
    SYNTAX  Counter
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
            "The total number of received UDP datagrams for
            which there was no application at the destination
            port."
    ::= { udp 2 }

udpInErrors OBJECT-TYPE
    SYNTAX  Counter
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
            "The number of received UDP datagrams that could
            not be delivered for reasons other than the lack
            of an application at the destination port."
    ::= { udp 3 }
```

```
udpOutDatagrams OBJECT-TYPE
    SYNTAX  Counter
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
            "The total number of UDP datagrams sent from this
            entity."
    ::= { udp 4 }

-- the UDP Listener table

-- The UDP listener table contains information about this
-- entity's UDP end-points on which a local application is
-- currently accepting datagrams.

udpTable OBJECT-TYPE
    SYNTAX  SEQUENCE OF UdpEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
            "A table containing UDP listener information."
    ::= { udp 5 }

udpEntry OBJECT-TYPE
    SYNTAX  UdpEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
            "Information about a particular current UDP
            listener."
    INDEX   { udpLocalAddress, udpLocalPort }
    ::= { udpTable 1 }

UdpEntry ::=
    SEQUENCE {
        udpLocalAddress
            IpAddress,
        udpLocalPort
            INTEGER (0..65535)
    }

udpLocalAddress OBJECT-TYPE
    SYNTAX  IpAddress
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
            "The local IP address for this UDP listener.  In
            the case of a UDP listener which is willing to
            accept datagrams for any IP interface associated
            with the node, the value 0.0.0.0 is used."
    ::= { udpEntry 1 }

udpLocalPort OBJECT-TYPE
    SYNTAX  INTEGER (0..65535)
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
            "The local port number for this UDP listener."
    ::= { udpEntry 2 }
```

Si resseguim cadascun dels objectes d'aquest grup ens queda una estructura com la següent:

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

Glossari

agent *m* En SNMP, mòdul programari que executa les funcions de gestió de xarxa que li demana el gestor. Gestor i agent es comuniquen via SNMP.

abstract syntax notation one *m* Llenguatge formal usat per a definir sintaxi. En el cas d'SNMP, la notació ASN.1 s'usa per a definir els objectes i el format del protocol.
sigla **ASN.1**

ASN.1 *m* Vegeu ***abstract syntax notation one***.

comunitat *f* En el context SNMP, relació entre agent i un grup de gestors que defineix característiques de seguretat. El concepte de comunitat és local i es defineix en l'agent.

base de dades d'informació *f* En el context SNMP, contenidor de la informació dels objectes per gestionar de forma estructurada on cada variable representa un recurs.
sigla **MIB**

estació de gestió *f* Mòdul programari que executa aplicacions de gestió per al control i monitorització d'elements de xarxa.
sin. **gestor**

gestor *m*
sin. **estació de gestió**

instància d'un objecte *f* Instància específica d'un tipus d'objecte que està vinculat a un valor específic.

MIB *f* Vegeu ***base de dades d'informació***.

objecte *m* Variable que representa algun recurs o altres aspectes del dispositiu gestionat.

remote network monitoring *m* Especificació de monitorització per a analitzar i monitoritzar protocols de xarxa.
sigla **RMON**

RMON *m* Vegeu ***remote network monitoring***.

sonda *f* En el context RMON, dispositiu de monitorització remota.

simple network management protocol *m* Col·lecció d'especificacions per a la gestió de xarxa que inclou el protocol, la definició de l'estructura de dades i altres conceptes associats.
sigla **SNMP**

SNMP *m* Vegeu ***simple network management protocol***.

telecomunication management network *m* Conceptualment, xarxa separada que interactua amb la xarxa de telecomunicacions a través de diversos punts. El concepte de TMN està definit per la recomanació M.3010.
sigla **TMN**

TMN *m* Vegeu ***telecomunication management network***.

trap *m* Missatge enviat per l'agent cap a l'estació de gestió sense que l'hagi sol·licitat.

Bibliografia

Mauro, D.R.; Schmidt, K. J. (2001). *Essential SNMP*. Sebastopol (EUA): O'Really.

(Disponible a http://www.unix.org.ua/orelly/networking_2ndEd/snmp)

Stalling, W. (1998). *SNMP, SNMPv2, SNMPv3 and RMON 1 and 2*. Reading (Massachussets): Addisson-Wesley.

Subramanian, M. (2000). *Network Management: Principles and Practice*. Reading (Massachussets): Addison-Wesley.

Udupa, D. (1999). *TMN Telecomunications Management Network*. McGraw-Hill