UNIVERSITY OF M'SILA FACULTY OF MATHEMATICS AND COMPUTER SCIENCE Computer Science Department

SNMP Protocol

Simple Network Management Protocol

Prepared by:

- **Ms. SAOUDI Lalia**

Master 2 RTIC: 2017/2018

Introduction

SNMP:

SNMP stands for Simple Network Management Protocol. It is a protocol, based on UDP, that allows administrators to monitor, remotely manage network devices and detect (via alarms) malfunctions and analyse faults.

Introduction

	Modèle *OSI*	Modèle *TCP/IP* (protocole)
7	Application	***SNMP***
6	Présentation	
5	Session	
4	Transport	*UDP*
3	Réseau	*IP*
2	Liaison	Interface réseau
1	Physique	

History SNMP.1

- In 1988, the need for a TCP-IP network management tool was felt:
- In 1991, version 1.0 of SNMP was published (RFC 1155, 1212, 1213, 1157).
- This version was widely criticised due to its lack of security:
  - Authentication is based on a password (community string field) transmitted in clear text over the network.

History: SNMP v2

- In 1993, SNMP v2
  - Introduced communication between managers
  - Improved object modelling (which remains close to SNMPv1)
  - Introduced a primitive limiting the number of grouped objects sent over the network.
  - And improved the structure of notification messages

SNMPv2C is the predominantly used experimental/intermediate version. However, it does not include the new security model of the time which is highly controversial.

History: SNMP v3

- Shifted the client-server model of SNMPv1/v2 to a peer-to-peer and modular model
- the concept of master/slave is replaced with that of entity
- All functionalities are decomposed into modules
- Provides real support in terms of security: confidentiality, authentication, message integrity

Definitions

- **Agent**

Software that allows a network component (switch, router...) to respond to requests from the management station.

- **Management station: (openview, ciscoworks...)**
  - Traces a profile of the network
  - Analyses MIB information
  - Creates graphs.

- **MIB: Management Information Base**

-Database containing the managed objects in each component of the network

SNMPv1/v2 Architecture – manager/agent

A manager

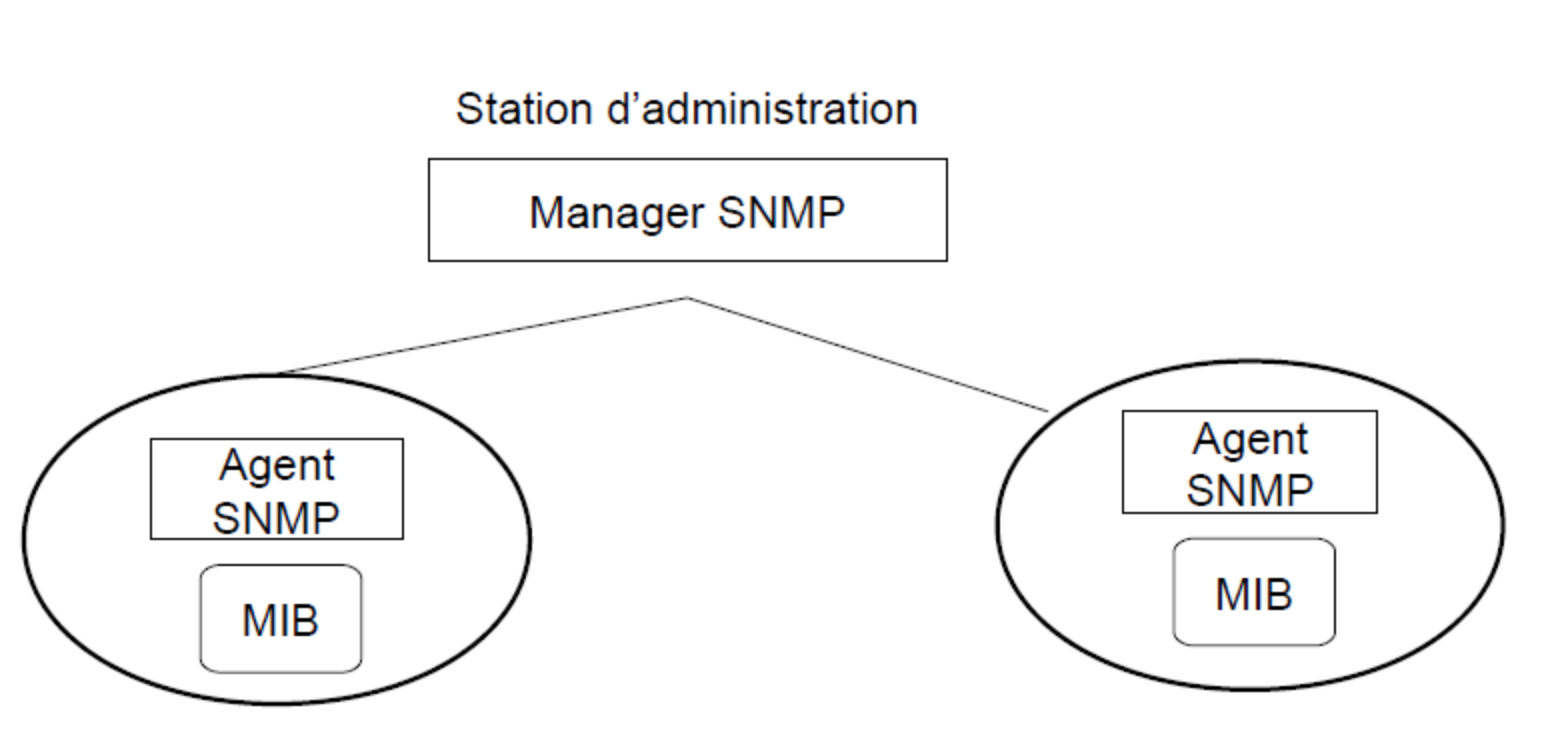
(software entity) centralises information from network devices via agents acting on behalf of the manager

The intelligence resides in the manager

An agent

agents are simple and lightweight with minimal impact on hosts

SNMPv1,v2 Architecture



SNMPv1,v2 Architecture

The manager:

- Sends commands (control) and receives corresponding responses: pull-type approach (question-answer).

- Receives unsolicited notifications. It is the manager's responsibility to issue commands to obtain more detail (push approach)

SNMPv1,v2 Architecture

The agent

- The agent provides access to a local MIB object which reflects the state of resources and the activity of the host.

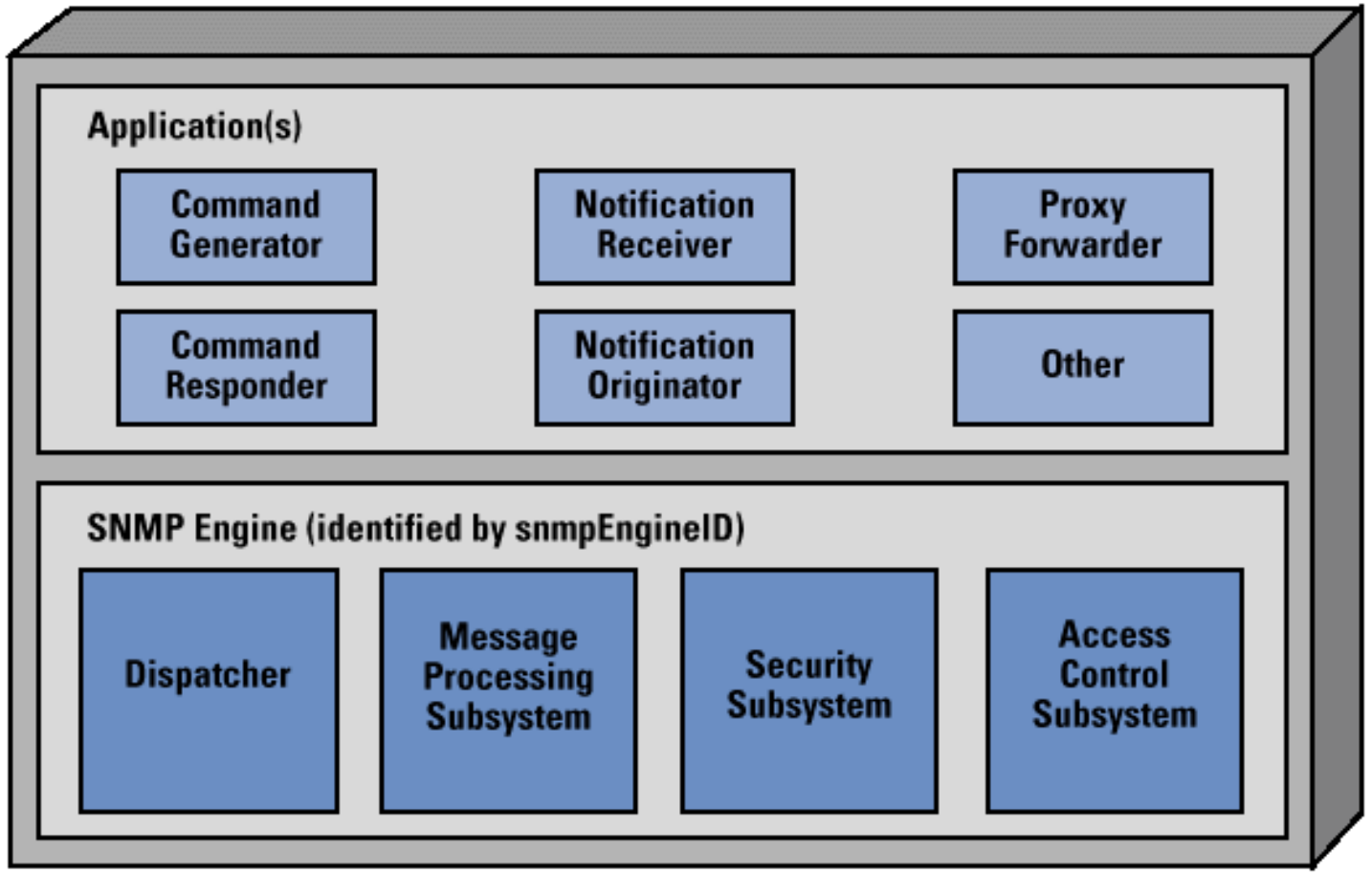
- The agent responds to the manager's commands by returning MIB values and assigning values

SNMPv3 Architecture

Introduces new terminology: we no longer talk of manager, agent but entity

- Represents a transition to a p2p and modular architecture.
- An SNMP entity is divided into two subsets:
  - The SNMP engine: consists of 4 subsystems.
  - Applications: 5 applications

SNMPv3 Architecture



SNMP requests

- Information search:
  - ✓GetRequest: searches for a variable on an agent.
  - ✓GetNextRequest: searches for the next variable.
  - ✓GetBulkRequest: searches for a group of variables
- Sending information
  - ✓Trap: incident detection
- Modification of values:
  - ✓SetRequest: allows changing the value of a variable of an agent.

SNMP responses

There is only one response.

It differs if there is an error or not.

- No error:

GetResponse: returns the desired value(s)

- In case of error:

GetResponse but accompanied by a NoSuchObject

Excerpt from RFC 1157 (SNMPv1)

Basic commands

– GET (manager -> agent)
• Requests a value
– GET-NEXT (manager -> agent)
• Retrieves the next value (list of values from a table)
– GET-RESPONSE (agent -> manager)
• Responds to GET/SET or error
– SET (manager -> agent)
• Sets a value or performs an action
– TRAP (agent -> manager)
• Spontaneous notification from the device (shutdown, temperature above threshold...)

SNMP relies on:

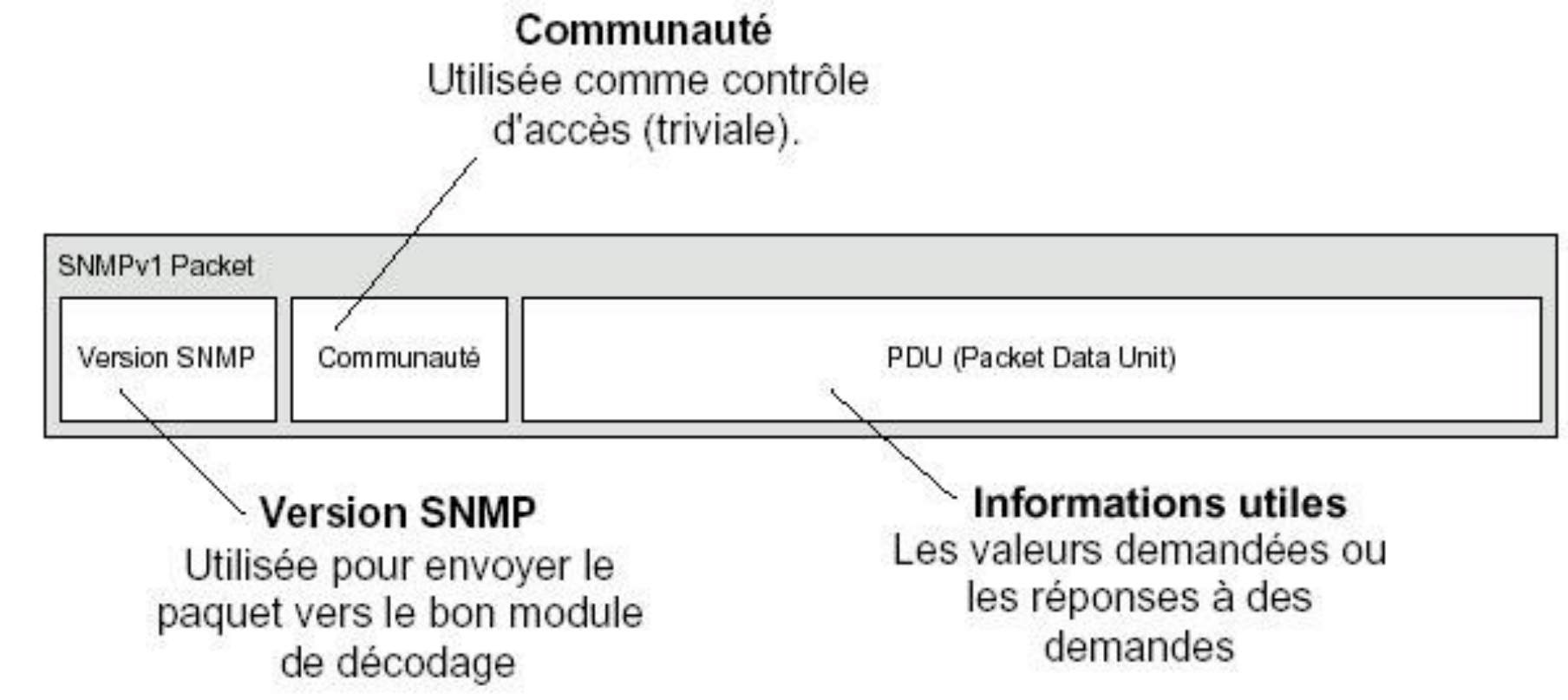
- UDP (unreliable, connectionless communication): an SNMP message is a packet transported via UDP on a port (standard port: 161 for commands and 162 for traps)

- ASN.1: (Abstract Syntax Notation One) is an international standard specifying a notation intended to describe data structures in the field of telecommunications and computer networks.

Excerpt from RFC 1157 (SNMPv1)

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

Format of an SNMPv1/2 message of a command

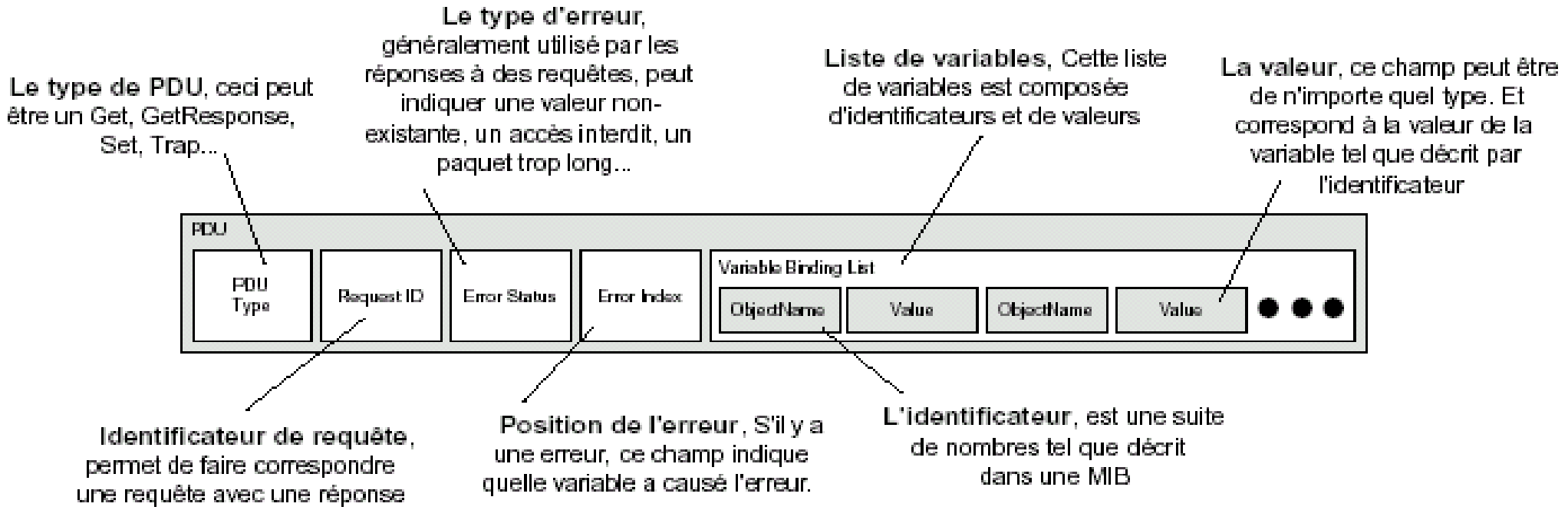


Format of an SNMPv1/2 message of a command

```
Message ::= SEQUENCE {
version -- version-1 for this RFC
INTEGER { version-1(0) },
community OCTET STRING, --
community name
data -- e.g., PDUs if trivial
ANY -- authentication is being used
}
```

Format of the PDU

The PDU consists of 5 fields including varbin List (for variable binding list) which is of constructed type (sequence of Varbin)



Excerpt from RFC 1157 (SNMPv1)

```
PDU ::= SEQUENCE {
request-id INTEGER,
error-status -- sometimes ignored
INTEGER { noError(0), tooBig(1),
noSuchName(2), badValue(3),
readOnly(4), genErr(5) },
error-index INTEGER, -- sometimes
ignored
variable-bindings -- values are
sometimes ignored VarBindList }
```

Format PDU

- **ID: request identifier (and consequently its response) assigned during the request**
- **Error: error code**
  - Placed by the SNMP agent if an error occurs,
  - Initialised to 0 in the request sent by the manager
- **Error index:**
  - index pointing to the first object that caused the error
  - Equals 0x00 if no error occurs
  - **Note: the GetBulk primitive uses the error and errorindex fields to store nonrepeater and maxrepetition values**

Format PDU

Varbind list:

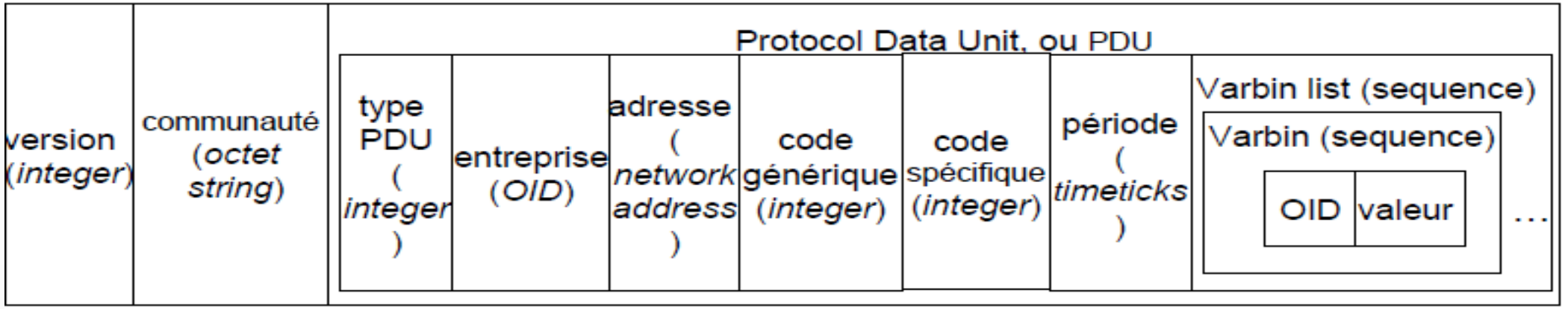
- Sequence consisting of the OID-value pair
- Represents a list of variables
- Depending on the type of message, the value differs.

- For example, for
  - SetRequest: the value corresponds to that of the specified OID
  - GetRequest: value is NULL
  - GetResponse the value corresponds to that of the specified SNMP agent's OID

Format PDU

Valeur du code d'erreur	explication
0x00	aucune erreur n'est survenue
0x01	réponse trop longue pour être transportée
0x02	le nom de l'objet demandé n'a pas été trouvé
0x03	le type de donnée de la requête ne « matche » pas celui se trouvant dans l'agent SNMP
0x04	le manager a tenté d'assigner une valeur à un paramètre accessible ne lecture seulement
0x05	0x05 – Erreur générale

Format PDU of a Trap



Enterprise: identification of the managed object generating the trap Address of the agent that generated the trap

the generic code (i.e. standardised) and specific (non-conventional) of the trap

Elapsed time since the last trap or reset

Management Information Base (MIB)

- A MIB is a collection of information
- organised hierarchically
- Accessed by a protocol (SNMP)
- A MIB comprises a set of managed objects (or MIB object)
- A MIB object represents a characteristic of the managed terminal

Management Information Base (MIB)

- A MIB is a data-oriented specification

defining the naming, type, format,
the actions concerning managed objects.

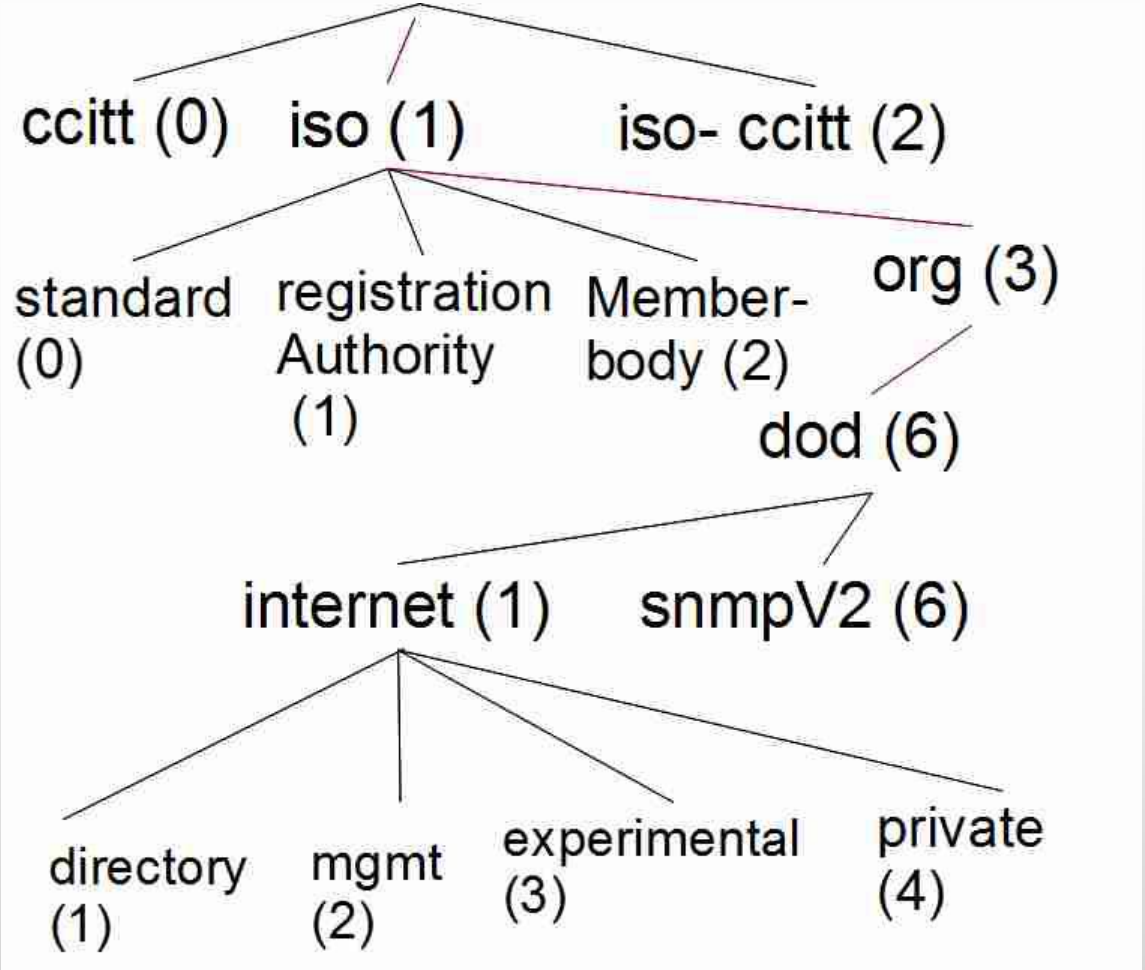
- The MIB forms an access interface to the underlying instrumentation; the objects

being accessed via the MIB (a sort of virtual database)

Hierarchical organisation of the MIB

- The managed objects through a MIB are organised in the form of a hierarchy
- This hierarchy (or tree) has a nameless root
  - Leaves corresponding to monitored objects
  - The different levels of the hierarchy are managed by different organisations (ISO, CCITT at level 1)

Hierarchical organisation of the MIB



Hierarchical organisation of the MIB

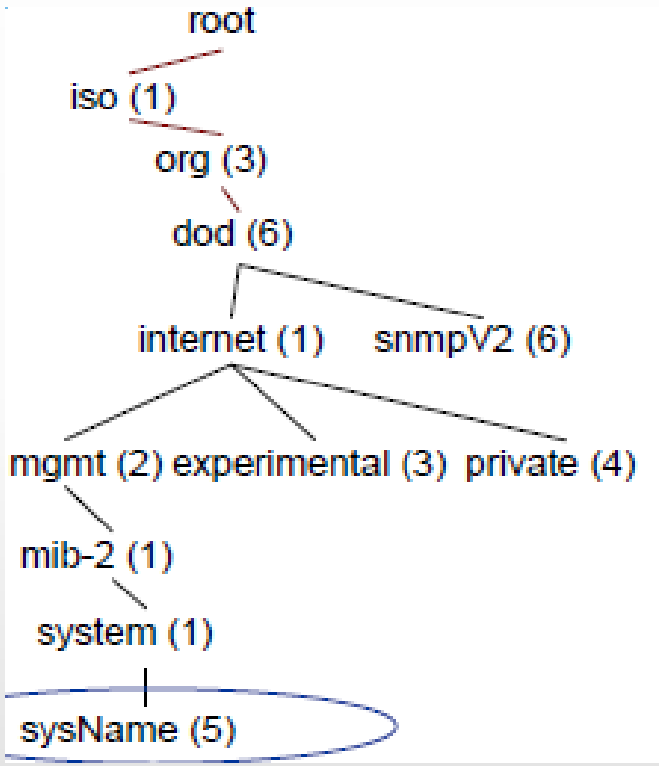
The hierarchy is organised into two parts:

- the part below 1.3.6 (dod)
  - is the only one used by SNMP
  - is intended for dod, the US Department of Defence which originally developed the Internet
- Private (4): vendors can define private branches including the MIB objects of their own products

Example - name of a machine

The coding of the OID follows the coding rules defined by ASN.1

- The object named sysName is considered
- Its OID is sysName OBJECT IDENTIFIER ::= { iso org(3)dod(6) internet(1) mgmt(2) mib-2(1) system(1) sysName(5)}
- Abbreviated form: .1.3.6.1.2.1.1.5



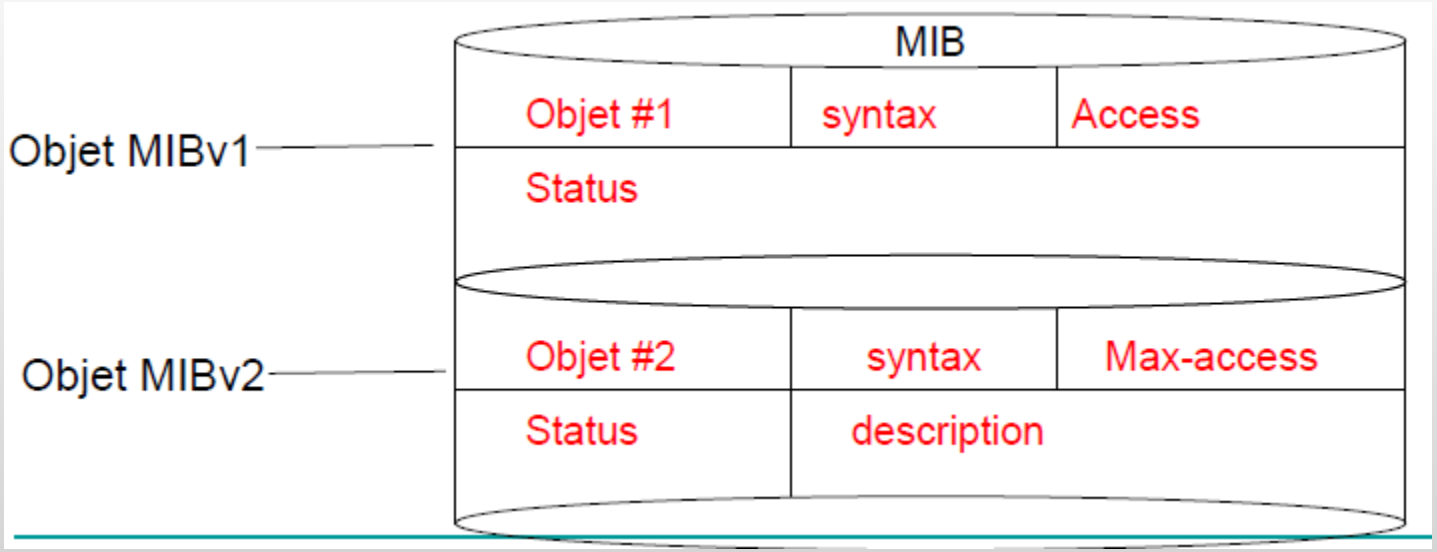
A MIB object

An object within a MIB is defined by
- its name
- Its type (SYNTAX)
- Its role (DESCRIPTION: string indicating the role of the object with SNMPv2)
- Its access rights (ACCESS with SMPv1 and MAXACCESS with SNMPv2), for example, read-only
Its state of (STATUS) (for example optional)

A MIB object

A MIB contains a set of objects

- The structure of objects evolves between versions 1 and 2 of SNMP (and SMI)
- With v1: 4 elements describe an object
- With v2: 5 elements describe an object



Access rights, SNMPv1 versus SNMPv2

Access right Description

read-only Read access

read-write Read and write access

write-only Write access

not-accessible Not accessible

read-create Access to creation (with SMIv2),

accessible-for-notify Access for trap (with SMIv2)

Status, SNMPv1 versus SNMPv2

Status Description
mandatory Mandatory presence (SNMPv1)
optional Optional presence (SNMPv1)
obsolete Is obsolete (SNMPv1 and SNMPv2)
deprecated is still supported but will become obsolete in the future or will be replaced by a more suitable object (SNMPv2)
current current (SNMPv2)

Example

sysUpTime OBJECT-TYPE
SYNTAX TimeTicks
ACCESS read-only
STATUS mandatory
DESCRIPTION
The time (in hundredths of a second) since the
network management portion of the system was last
re-initialised.
::= { system 3 }

sysUpTime OBJECT-TYPE
Defines the object sysUpTime.
SYNTAX TimeTicks
Object of type TimeTicks. The object types are specified in the previously mentioned SMI.
ACCESS read-only
This object can only be read by SNMP (get request); it cannot be modified (set request).
STATUS mandatory
This object must be implemented on any SNMP agent.
DESCRIPTION
Description of the object
::= { system 3 }
The sysUpTime object constitutes the third branch of the subtree of the system object group

Example

```
ACCELANCE-MIB DEFINITIONS ::= BEGIN
          …
accelance MODULE-IDENTITY
          …
    ::= { enterprises 9697 }

general OBJECT IDENTIFIER ::= { accelance 1 }
          …
release  OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
          "Type d'OS utilisé"
    ::= { general 1 }
```

Identification d'un fichier ASN.1

Rattachement de la branche *Accelance* à la branche *enterprises* via l'OID 9697

Définition de la variable *realease*

&

Rattachement de celle-ci à la branche *general*

- We can now access the release variable as follows:

iso.dod.internet.private.enterprises.accelance.general.release.0

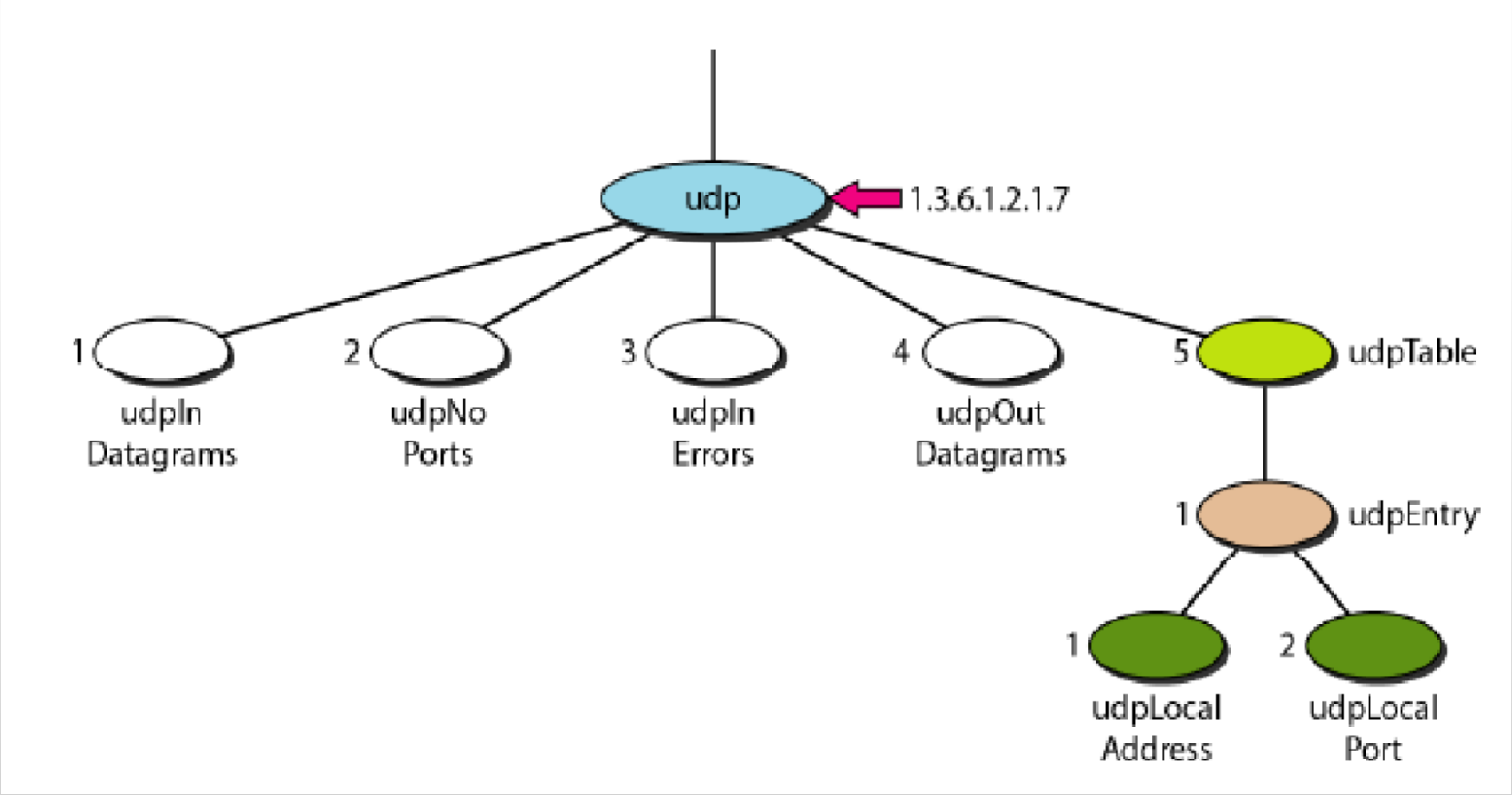
Groups in MIB-2

The objects in MIB_2 are categorised under 10 different groups:

Group Description

System: Sys Name, location, lifetime,,,
Interfaces: If Interface number, Physical, &Ip Address, Traffic
AT Address Translation, information about the ARP table
IP Statistics on IP packets, routing table
ICMP ICMP statistics
TCP connection table, number of ports, number of packets sent and received
UDP general information related to UDP, such as the number of ports, number of packets sent and received. EGP Traffic statistics EGP (historical usage)
Reserved Transmission
SNMP Traffic Statistics

UDP Groups in MIB-2



SMI

At the lowest level, SNMP variables
are defined as

- individual objects to describe managed objects
- as notifications

Related objects are grouped together in modules
Thus, an SNMP agent only manages the groups and modules it needs

Conclusion

- SNMP is the de facto standard for network management (IP)
- SNMP is widely implemented (routers, bridges, telecommunications equipment) by manufacturers/vendors,
- SNMP only meets some of the functional requirements for network management

- Course Questions 2
- Q1: What is the meaning of SNMP,
- Q2: What is the difference between its three versions?
- Q3: What methods are called by the SNMP agent and the methods called by its manager?
- Q4: Why is the SNMP protocol the most widely used?