Network Management:

Basics, Standards and Evolution toward Distributed, Intelligent and Cost-effective Architectures

Raouf Boutaba
School of Computer Science
University of Waterloo
Waterloo, Ontario, N2L 3G1, CANADA

Phone: +1 519 888 4820
Email: rboutaba@bbcr.uwaterloo.ca
Web: http://bbcr.cs.uwaterloo.ca/~rboutaba

Course Outline

- **Course Objective and Motivation**
- **Simple Network Management**
- **Remote Network Monitoring in TCP/IP Networks**
- **Advanced Management of TCP/IP Networks**
- **Management of Telecommunication Networks**
- **Internet Technologies for Converged Networks Management**

Course - Objectives

- ✓ *Appreciate the need for interoperable network management*
- ✓ *Understand general concepts and architecture behind standards based network management*
- ✓ *Understand concepts and terminology associated with SNMP and TMN*
- ✓ *Appreciate network management as a typical distributed application*
- ✓ *Get a feeling of current trends in network management technologies*
- ✓ *Understand Advanced Information Processing Techniques such as Distributed Object Technologies, Software Agents and Internet Technologies used for network management*

Why is network management needed ?

In a perfect world, networks would not need management - they would just run themselves.

However…

- *Parts tend to break*
- *Changes are made*
- *Somebody has to pay*
- *Performance does not meet expectations*
- *Abuse happens*

What is network management ?

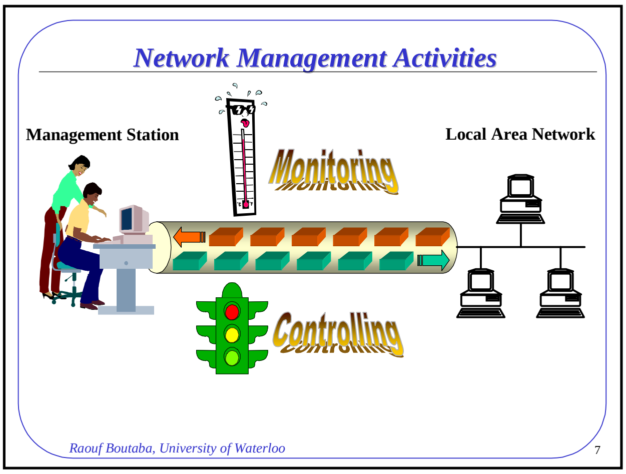
Monitoring/controlling the network & Planning the network evolution.

Management Functional Areas ("FCAPS"):

- **Fault Management**
  Maintain error logs, handle fault notifications, trace faults, diagnostic tests, correct faults,
- **Configuration Management**
  Record configuration, record changes, identify components, init/stop system, change parameters,
- **Accounting Management**
  Establish charges, identify utilization costs, billing, …
- **Performance Management**
  Optimize QoS (Quality of Service), detect changes in performances, collect statistics, …
- **Security Management**
  key management (authorization, encryption & authentication), firewalls, security logs, ...

Module 1 - Objectives

- ➢ *describe what is meant by network management*
- ➢ *explain the concepts of network management*
- ➢ *outline the classes of data collected from monitoring a network*
- ➢ *outline the standards for network management, here the IETF*
- ➢ *describe how a standardized form of network management is implemented*



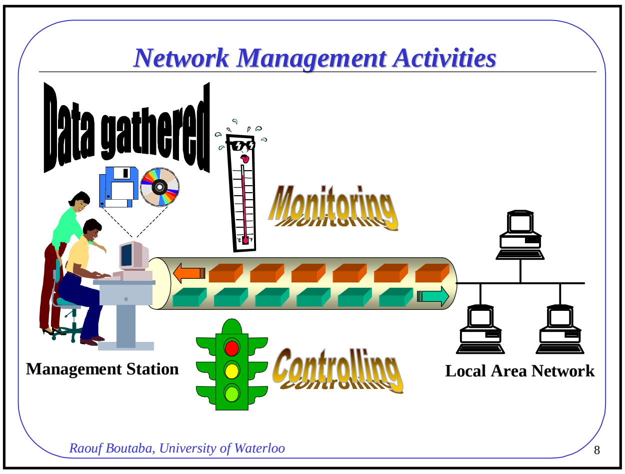
Network Management Activities

Management Station

Monitoring

Local Area Network

Controlling



Network Management Activities

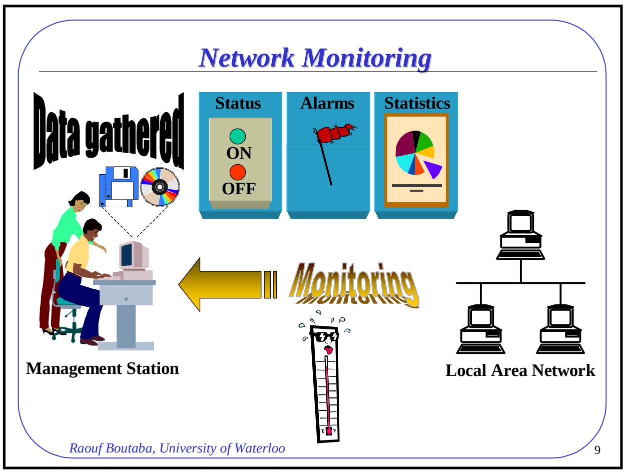
Data gathered

Monitoring

Management Station

Controlling

Local Area Network



Network Monitoring

Data gathered

Status: ON, OFF

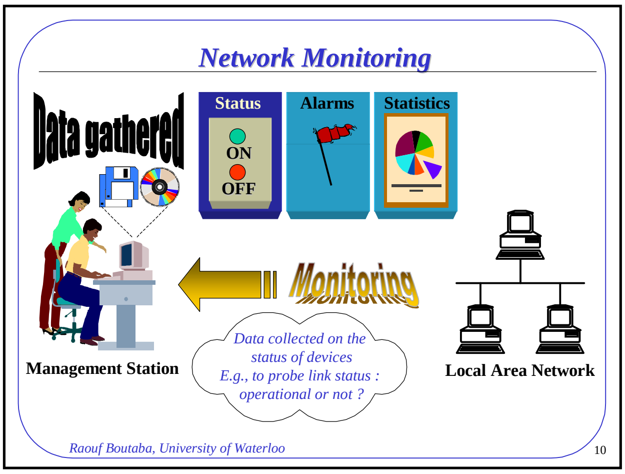
Alarms

Statistics

Monitoring

Management Station

Local Area Network



Network Monitoring

Data gathered

Status: ON, OFF

Alarms

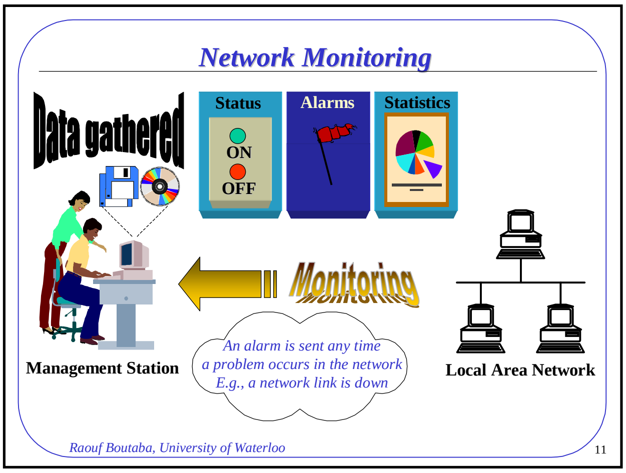
Statistics

Monitoring

Management Station

Local Area Network

Data collected on the status of devices
E.g., to probe link status : operational or not ?



Network Monitoring

Data gathered

Status: ON, OFF

Alarms

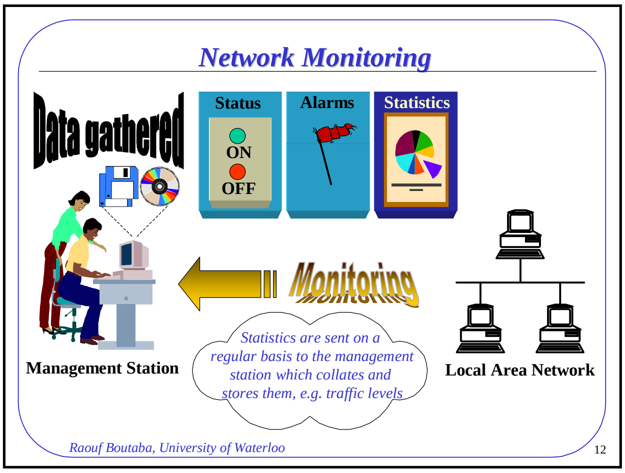
Statistics

Monitoring

Management Station

Local Area Network

An alarm is sent any time a problem occurs in the network
E.g., a network link is down



Network Monitoring

Data gathered

Status: ON, OFF

Alarms

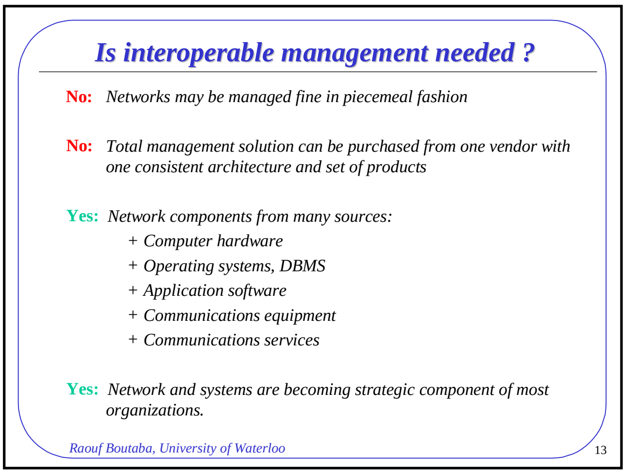
Statistics

Monitoring

Management Station

Local Area Network

Statistics are sent on a regular basis to the management station which collates and stores them, e.g. traffic levels



Is interoperable management needed ?

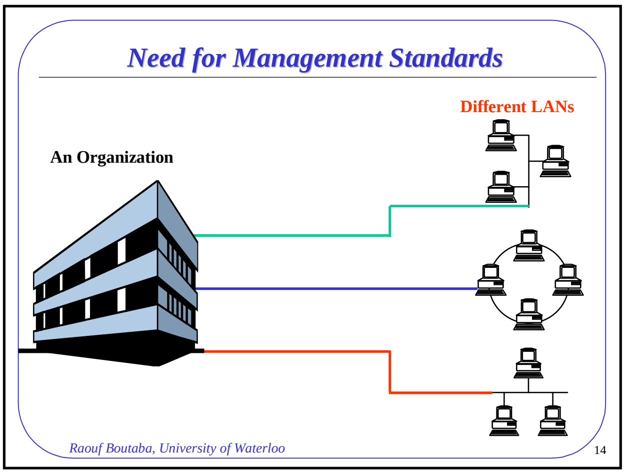
No: *Networks may be managed fine in piecemeal fashion*

No: *Total management solution can be purchased from one vendor with one consistent architecture and set of products*

Yes: *Network components from many sources:*
- + *Computer hardware*
- + *Operating systems, DBMS*
- + *Application software*
- + *Communications equipment*
- + *Communications services*

Yes: *Network and systems are becoming strategic component of most organizations.*

Raouf Boutaba, University of Waterloo 13

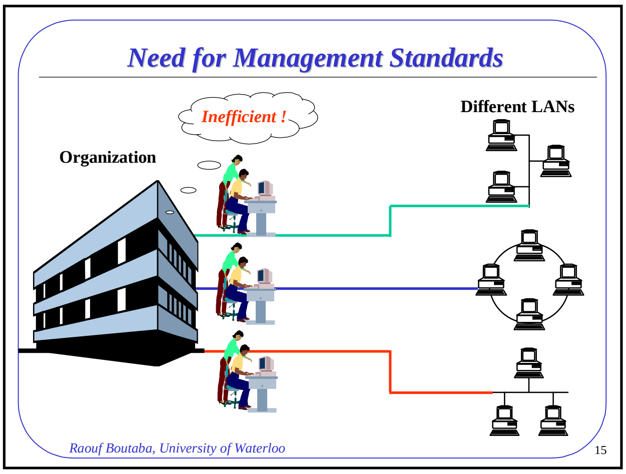


Need for Management Standards

Different LANs

An Organization

Raouf Boutaba, University of Waterloo 14



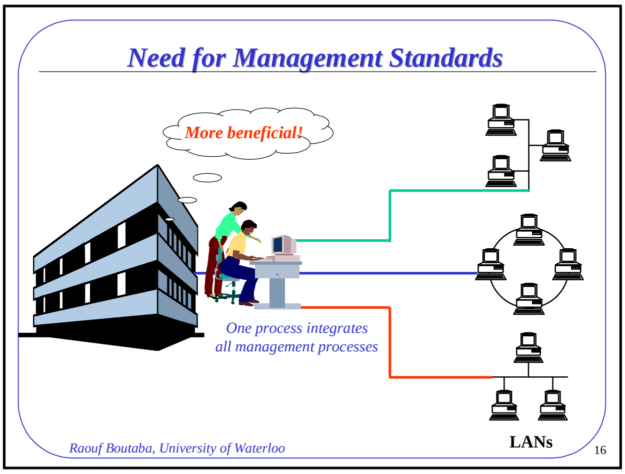
Need for Management Standards

Inefficient !

Different LANs

Organization

Raouf Boutaba, University of Waterloo 15



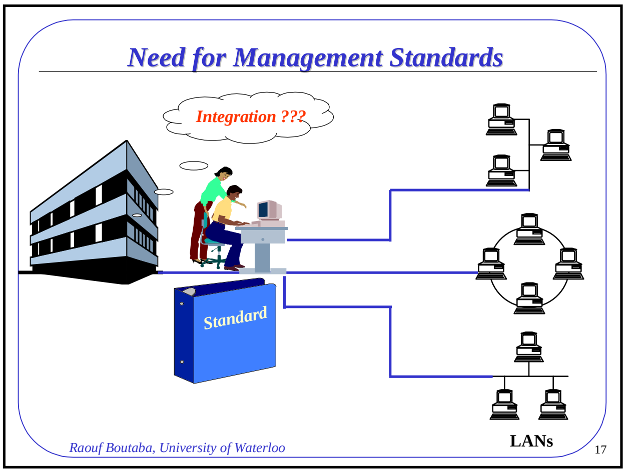
Need for Management Standards

More beneficial!

One process integrates all management processes

LANs

Raouf Boutaba, University of Waterloo 16



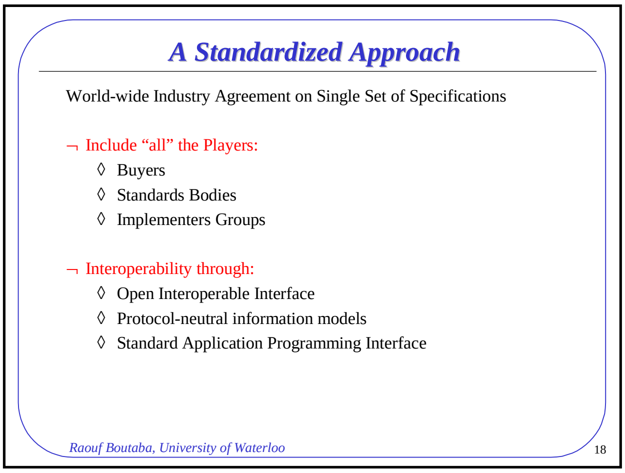
Need for Management Standards

Integration ???

Standard

LANs

Raouf Boutaba, University of Waterloo 17



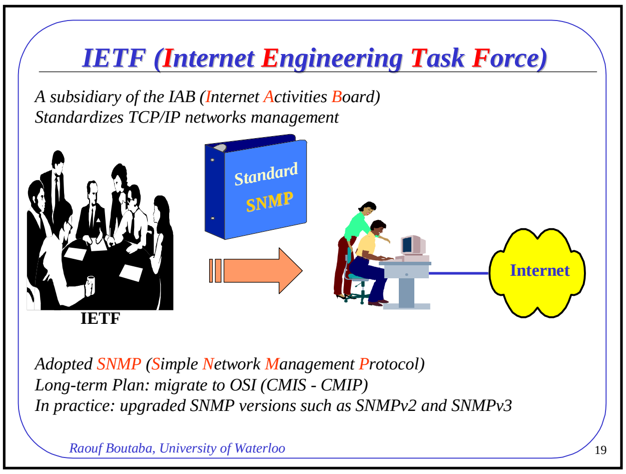
A Standardized Approach

World-wide Industry Agreement on Single Set of Specifications

¬ Include “all” the Players:
- ◊ Buyers
- ◊ Standards Bodies
- ◊ Implementers Groups

¬ Interoperability through:
- ◊ Open Interoperable Interface
- ◊ Protocol-neutral information models
- ◊ Standard Application Programming Interface

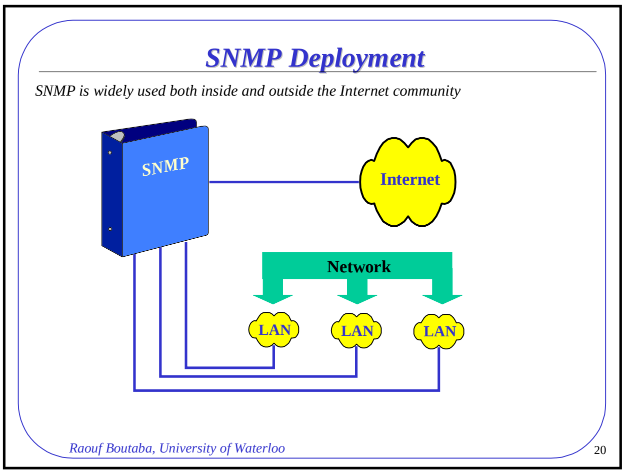
Raouf Boutaba, University of Waterloo 18



IETF (Internet Engineering Task Force)

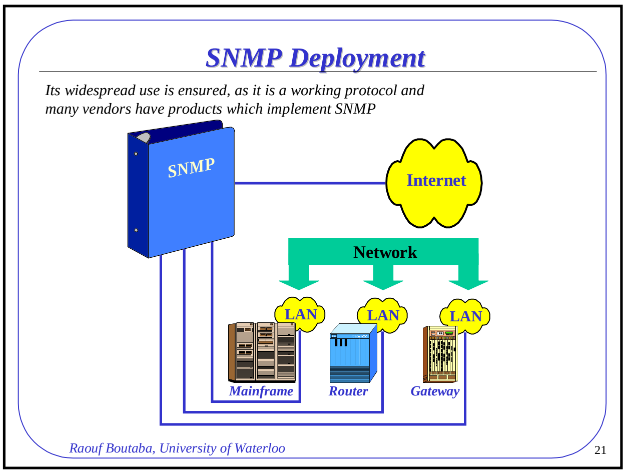
A subsidiary of the IAB (Internet Activities Board)
Standardizes TCP/IP networks management

Adopted SNMP (Simple Network Management Protocol)
Long-term Plan: migrate to OSI (CMIS - CMIP)
In practice: upgraded SNMP versions such as SNMPv2 and SNMPv3



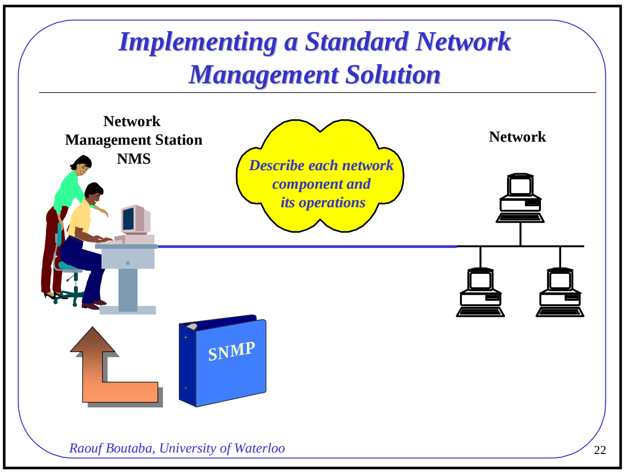
SNMP Deployment

SNMP is widely used both inside and outside the Internet community

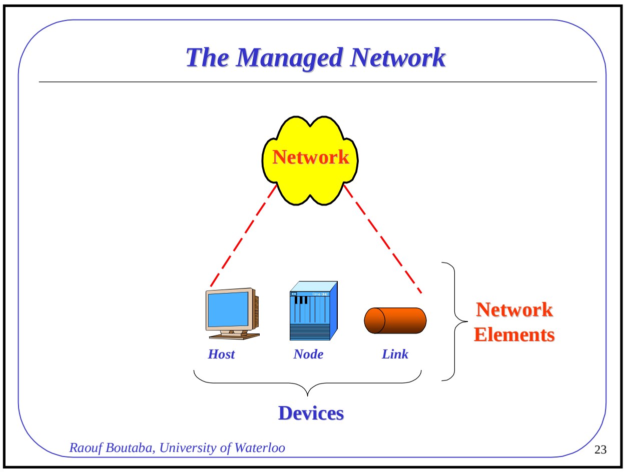


SNMP Deployment

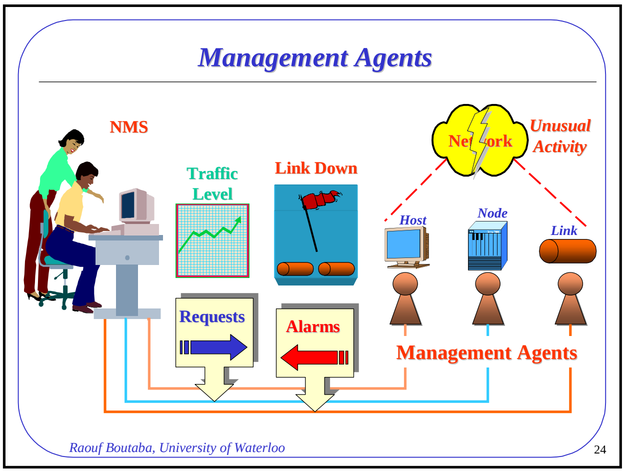
Its widespread use is ensured, as it is a working protocol and many vendors have products which implement SNMP



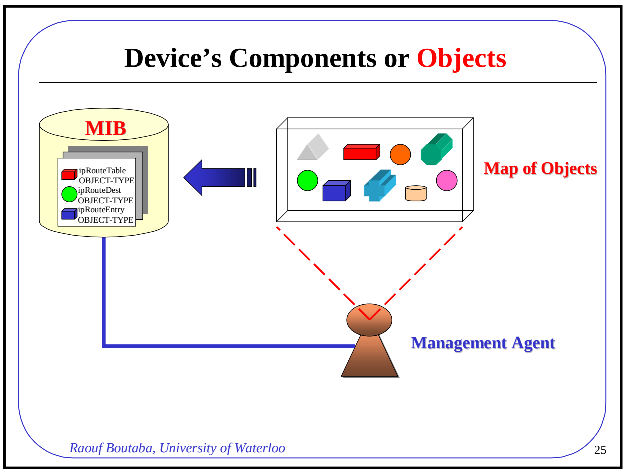
Implementing a Standard Network Management Solution



The Managed Network



Management Agents



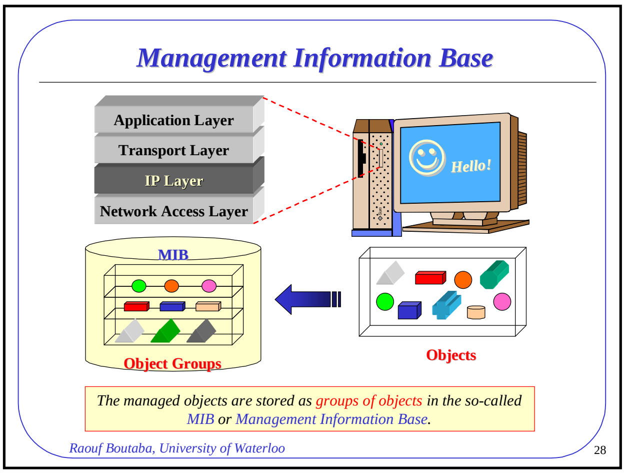
Device's Components or Objects

Summary so far

- *Network management is the activity of monitoring the network and using the data collected to control it.*
- *The monitoring data can be : Current status; Alarms; Statistics.*
- *The Simple Network Management Protocol has been adopted by the IETF as the standard protocol for managing Internet networks .*
- *A Managed device, known as a network element, is represented by a management agent which communicates with the NMS on behalf of the device.*
- *The Management agent accesses the associated device's components, called managed objects, to obtain monitoring data or to perform the MNS control actions*

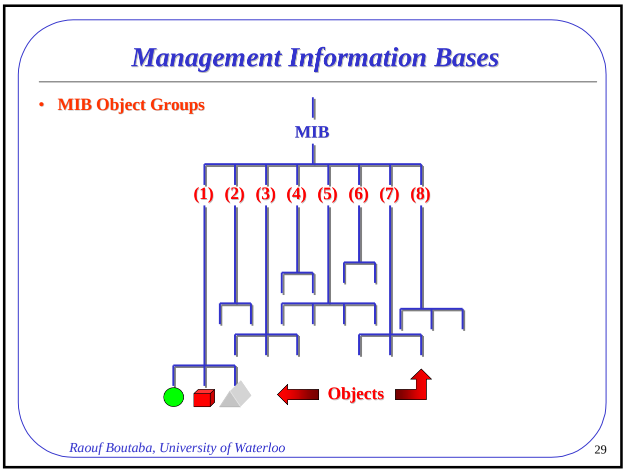
Management Information Bases

- **Standard MIB Structure**
- **MIB Objects Description**
- **MIB Objects Detailed Description**



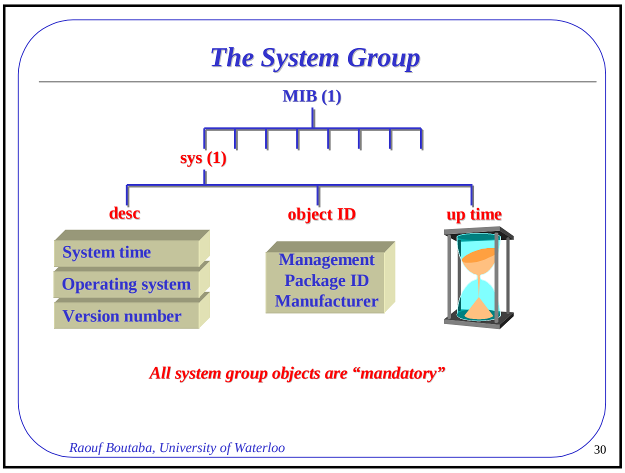
Management Information Base

The managed objects are stored as groups of objects in the so-called MIB or Management Information Base.



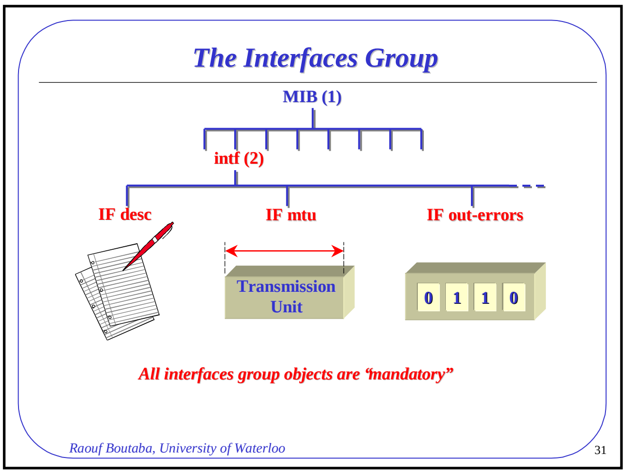
Management Information Bases

- **MIB Object Groups**



The System Group

All system group objects are "mandatory"



The Interfaces Group

MIB (1)

intf (2)

IF desc

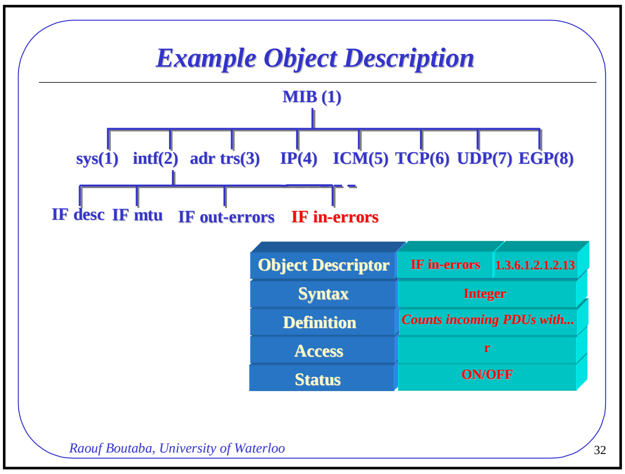
IF mtu

IF out-errors

Transmission Unit

0 1 1 0

All interfaces group objects are "mandatory"



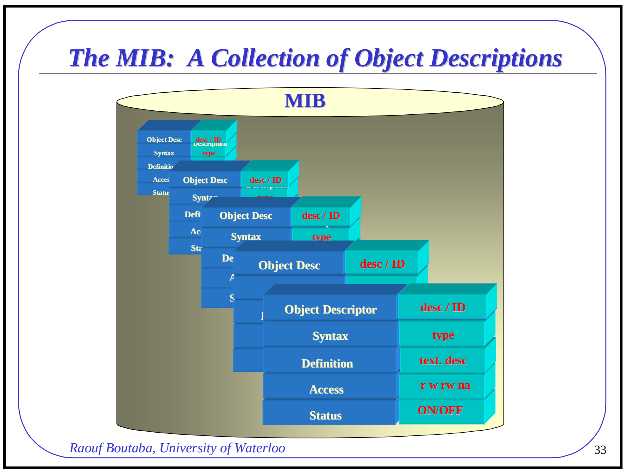
Example Object Description

MIB (1)

sys(1) intf(2) adr trs(3) IP(4) ICM(5) TCP(6) UDP(7) EGP(8)

IF desc IF mtu IF out-errors IF in-errors

Object Descriptor	IF in-errors	1.3.6.1.2.1.2.13
Syntax	Integer	
Definition	*Counts incoming PDUs with...*	
Access	r	
Status	ON/OFF	



The MIB: A Collection of Object Descriptions

MIB

Object Desc | desc / ID

Syntax | type

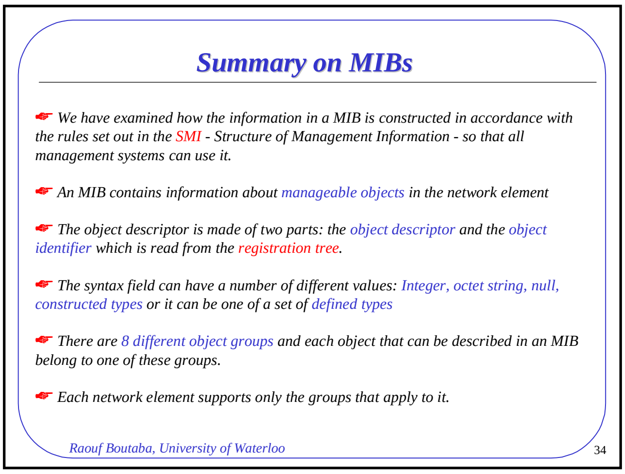
Object Descriptor | desc / ID

Syntax | type

Definition | text. desc

Access | r w rw na

Status | ON/OFF



Summary on MIBs

☞ We have examined how the information in a MIB is constructed in accordance with the rules set out in the SMI - Structure of Management Information - so that all management systems can use it.

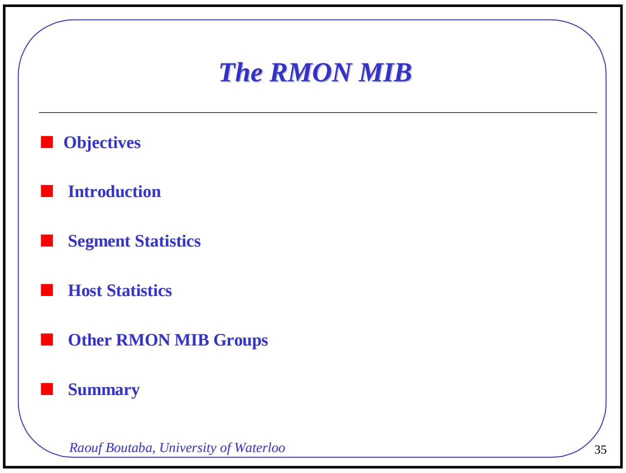
☞ An MIB contains information about manageable objects in the network element

☞ The object descriptor is made of two parts: the object descriptor and the object identifier which is read from the registration tree.

☞ The syntax field can have a number of different values: Integer, octet string, null, constructed types or it can be one of a set of defined types

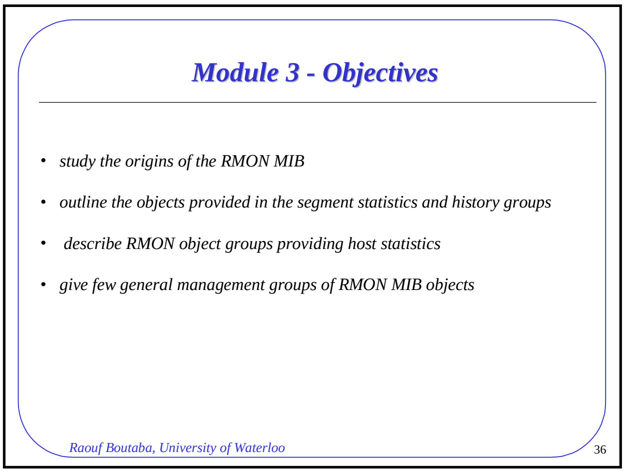
☞ There are 8 different object groups and each object that can be described in an MIB belong to one of these groups.

☞ Each network element supports only the groups that apply to it.



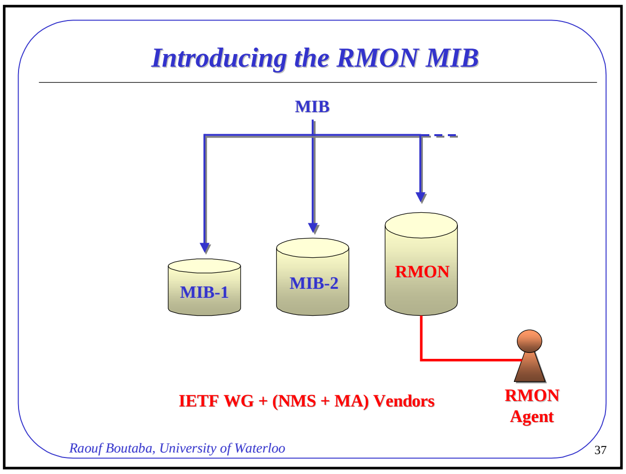
The RMON MIB

- Objectives
- Introduction
- Segment Statistics
- Host Statistics
- Other RMON MIB Groups
- Summary



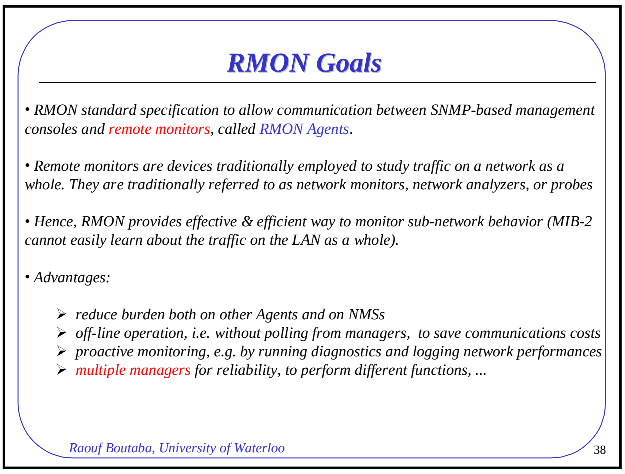
Module 3 - Objectives

- *study the origins of the RMON MIB*
- *outline the objects provided in the segment statistics and history groups*
- *describe RMON object groups providing host statistics*
- *give few general management groups of RMON MIB objects*



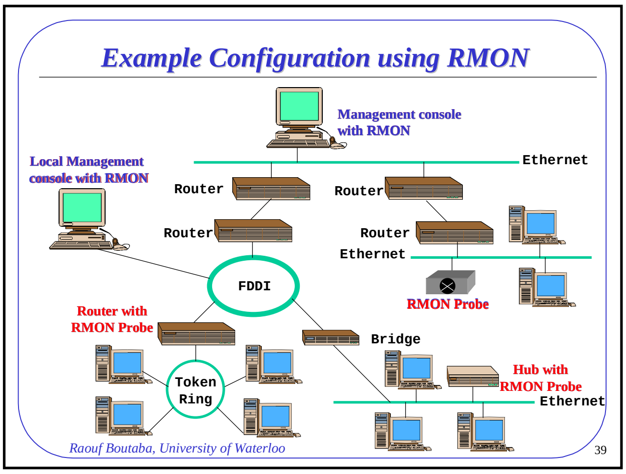
Introducing the RMON MIB

IETF WG + (NMS + MA) Vendors

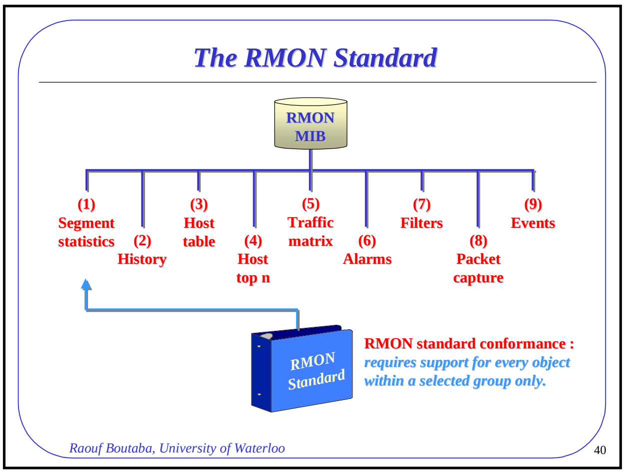


RMON Goals

- RMON standard specification to allow communication between SNMP-based management consoles and remote monitors, called RMON Agents.
- Remote monitors are devices traditionally employed to study traffic on a network as a whole. They are traditionally referred to as network monitors, network analyzers, or probes
- Hence, RMON provides effective & efficient way to monitor sub-network behavior (MIB-2 cannot easily learn about the traffic on the LAN as a whole).
- Advantages:
  - ➢ reduce burden both on other Agents and on NMSs
  - ➢ off-line operation, i.e. without polling from managers, to save communications costs
  - ➢ proactive monitoring, e.g. by running diagnostics and logging network performances
  - ➢ multiple managers for reliability, to perform different functions, ...

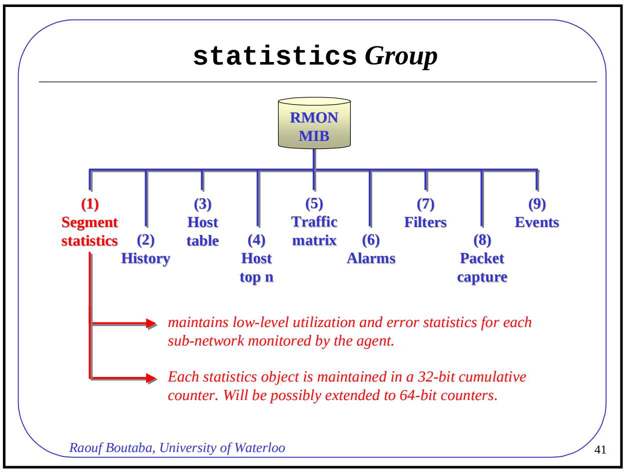


Example Configuration using RMON



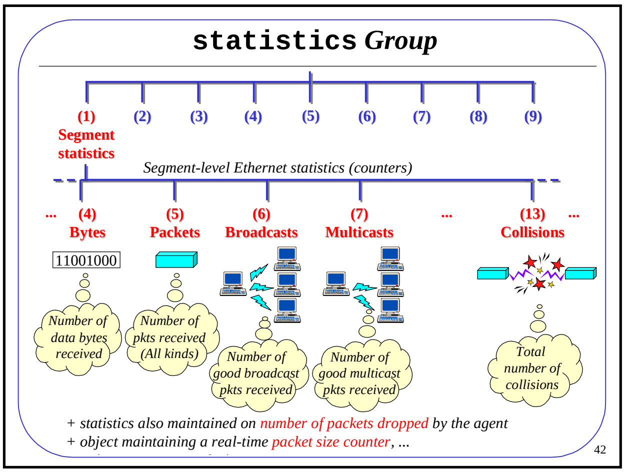
The RMON Standard

RMON standard conformance : requires support for every object within a selected group only.



statistics Group

- maintains low-level utilization and error statistics for each sub-network monitored by the agent.
- Each statistics object is maintained in a 32-bit cumulative counter. Will be possibly extended to 64-bit counters.



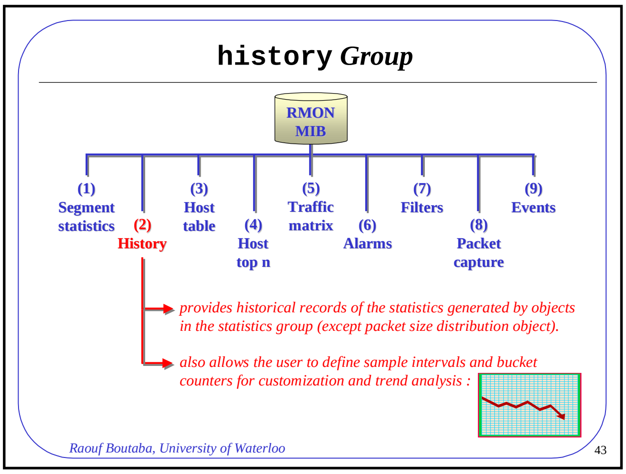
statistics Group

(1) Segment statistics
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)

Segment-level Ethernet statistics (counters)

...
(4) Bytes
(5) Packets
(6) Broadcasts
(7) Multicasts
...
(13) Collisions
...
11001000
Number of data bytes received
Number of pkts received (All kinds)
Number of good broadcast pkts received
Number of good multicast pkts received
Total number of collisions

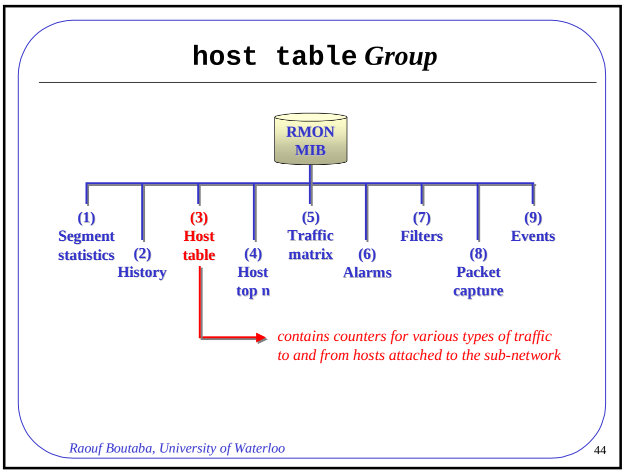
+ statistics also maintained on number of packets dropped by the agent
+ object maintaining a real-time packet size counter, ...



`history` *Group*

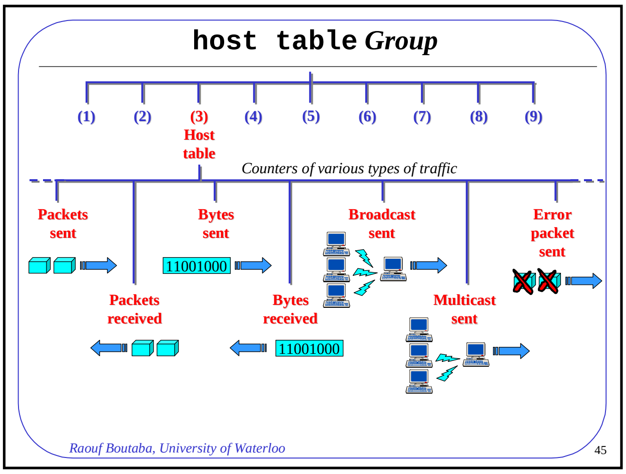
provides historical records of the statistics generated by objects in the statistics group (except packet size distribution object).

also allows the user to define sample intervals and bucket counters for customization and trend analysis :

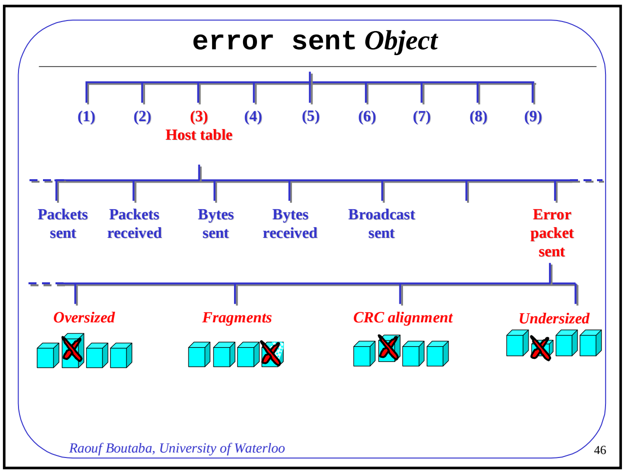


`host table` *Group*

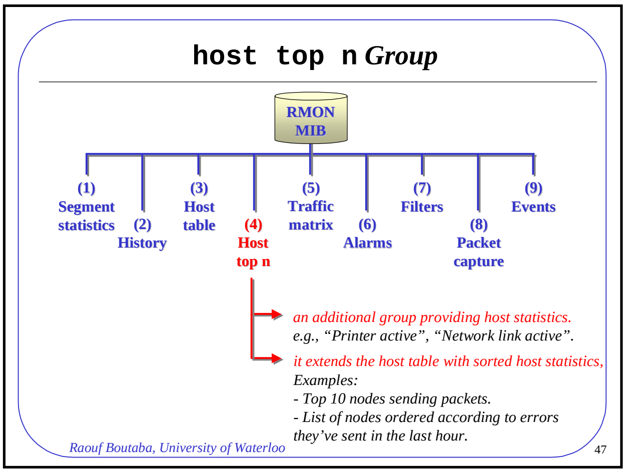
contains counters for various types of traffic to and from hosts attached to the sub-network



`host table` *Group*



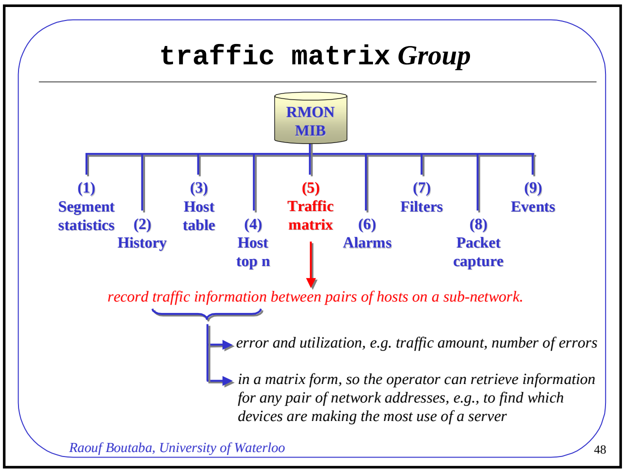
`error sent` *Object*



`host top n` *Group*

an additional group providing host statistics. e.g., "Printer active", "Network link active".

it extends the host table with sorted host statistics, Examples:
- Top 10 nodes sending packets.
- List of nodes ordered according to errors they've sent in the last hour.

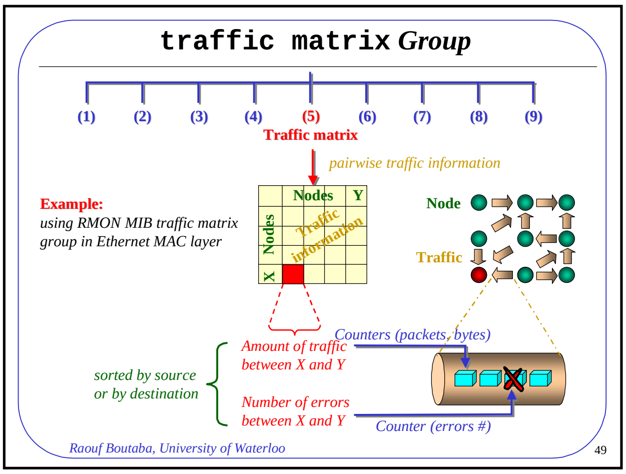


`traffic matrix` *Group*

record traffic information between pairs of hosts on a sub-network.

error and utilization, e.g. traffic amount, number of errors

in a matrix form, so the operator can retrieve information for any pair of network addresses, e.g., to find which devices are making the most use of a server



`traffic matrix` *Group*

Traffic matrix

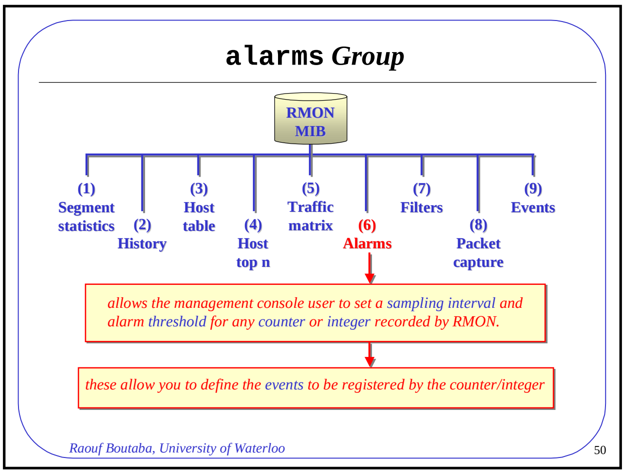
pairwise traffic information

Example:
using RMON MIB traffic matrix group in Ethernet MAC layer

sorted by source or by destination

Amount of traffic between X and Y

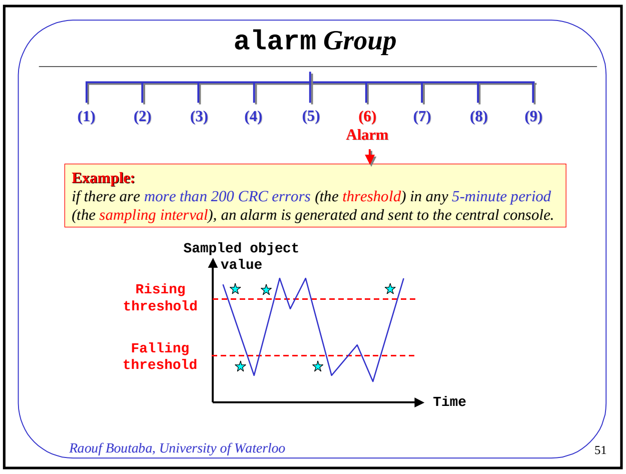
Number of errors between X and Y



`alarms` *Group*

allows the management console user to set a sampling interval and alarm threshold for any counter or integer recorded by RMON.

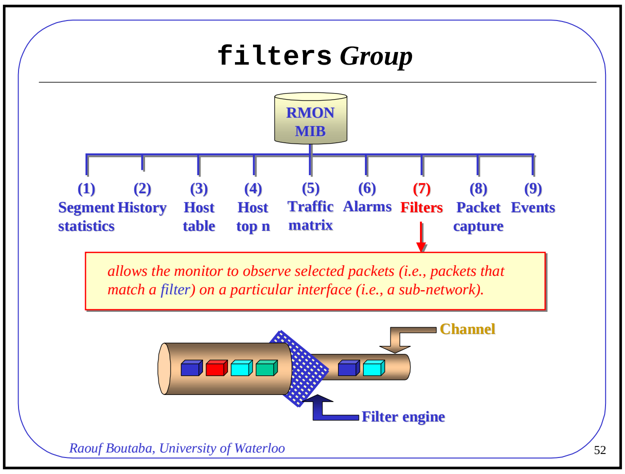
these allow you to define the events to be registered by the counter/integer



`alarm` *Group*

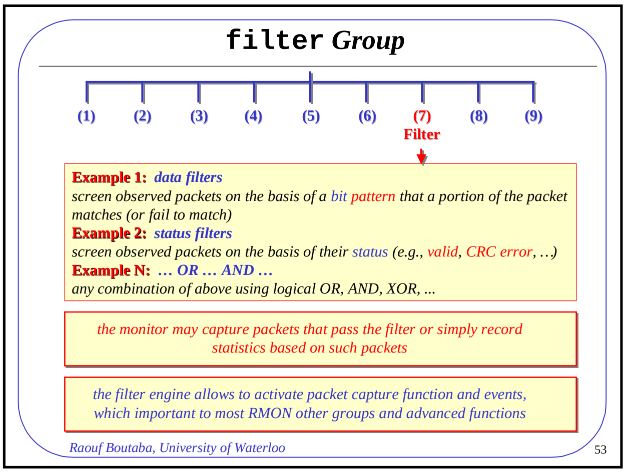
Alarm

Example:
if there are more than 200 CRC errors (the threshold) in any 5-minute period (the sampling interval), an alarm is generated and sent to the central console.



`filters` *Group*

allows the monitor to observe selected packets (i.e., packets that match a filter) on a particular interface (i.e., a sub-network).



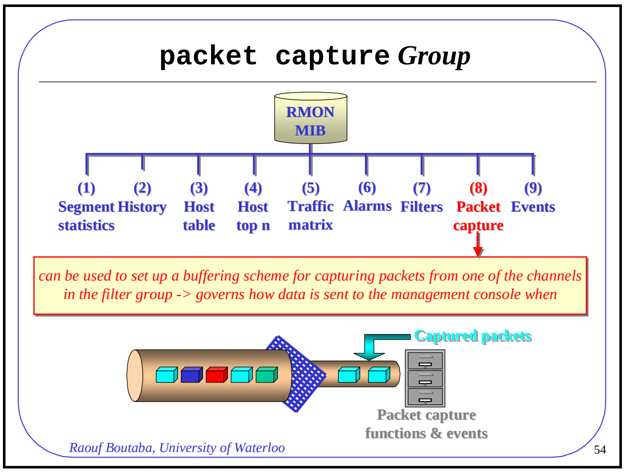
`filter` *Group*

Filter

Example 1: *data filters*
screen observed packets on the basis of a bit pattern that a portion of the packet matches (or fail to match)
Example 2: *status filters*
screen observed packets on the basis of their status (e.g., valid, CRC error, …)
Example N: *… OR … AND …*
any combination of above using logical OR, AND, XOR, ...

the monitor may capture packets that pass the filter or simply record statistics based on such packets

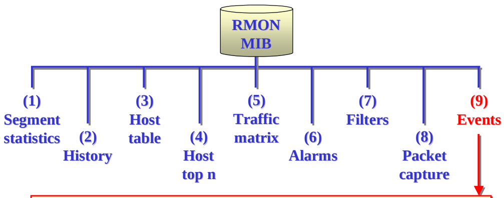
the filter engine allows to activate packet capture function and events, which important to most RMON other groups and advanced functions



`packet capture` *Group*

can be used to set up a buffering scheme for capturing packets from one of the channels in the filter group -> governs how data is sent to the management console when

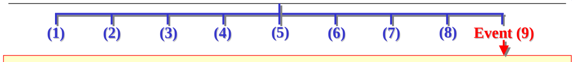
`event` *Group*



supports the definition of events and gives a table of all events generated by the RMON probe.

An event is triggered by a condition located elsewhere in the MIB, and an event can trigger an action defined elsewhere in the MIB

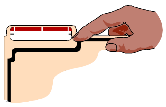
`event` *Group*

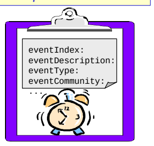


*An **event** may cause information to be **logged** in this group and may cause an SNMP **trap** message to be issued.*

A probe log includes the time at which each event occurred as well as a description of the event

Traps are error trapping contained in SNMP messages from agent to NMS





Conditions of an event to occur are defined in other RMON groups
E.g.: *alarm group can define threshold event referenced by indexing in* `eventTable`
E.g.: *filter group can reference an event that will occur when a packet is captured*

Summary on RMON MIB

☑ *The RMON MIB was developed by the IETF. It consists of nine groups of objects.*

☑ *Compliance with the RMON MIB standard only requires support for one object from within each group.*

☑ *Segment statistics provides segment-level Ethernet statistics on packets, bytes, broadcasts, multicasts, collisions and packet size distribution. The history group provides customized historical data on most of these.*

☑ *The RMON MIB provides the host table and host top n groups containing objects for a range of host statistics.*

☑ *The RMON MIB also provides:*
- *a traffic matrix group for statistics on traffic between pairs of nodes*
- *an alarms group for setting thresholds and sampling intervals*
- *a filters group for activating packet capture functions and events*
- *a packet capture group for capturing LAN packets*
- *an event group for creating log entries and traps*

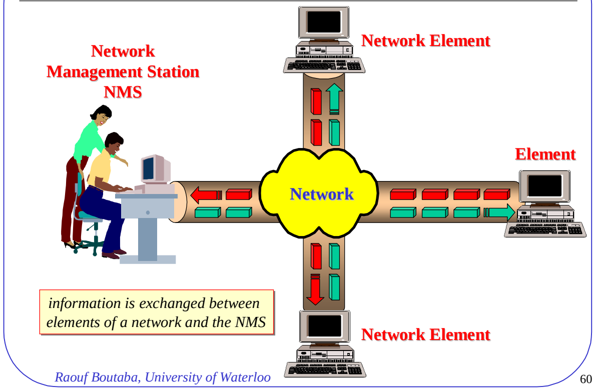
The Simple Network Management Protocol

- **Objectives**
- **SNMP protocol operation**
- **SNMP messages**
- **SNMP PDUs**
- **Structure of SNMP PDUs**
- **Summary**

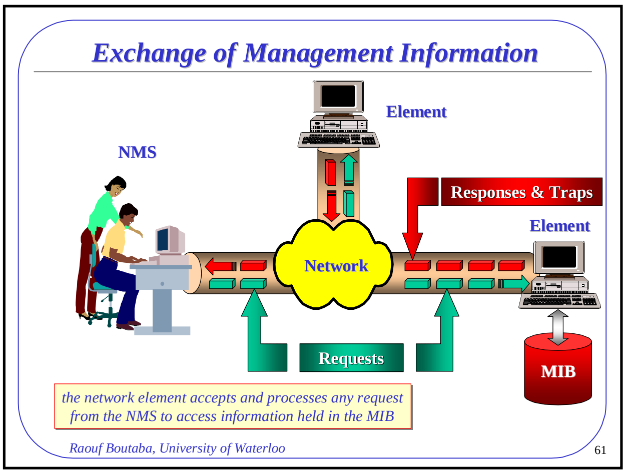
Objectives

- *Explain the role SNMP plays in exchanging network management information between the NMS and the network elements*
- *Describe the structure of SNMP messages and outline SNMP PDU types*
- *Describe the structure of SNMP GET, SET and TRAP PDUs*

Exchange of Management Information

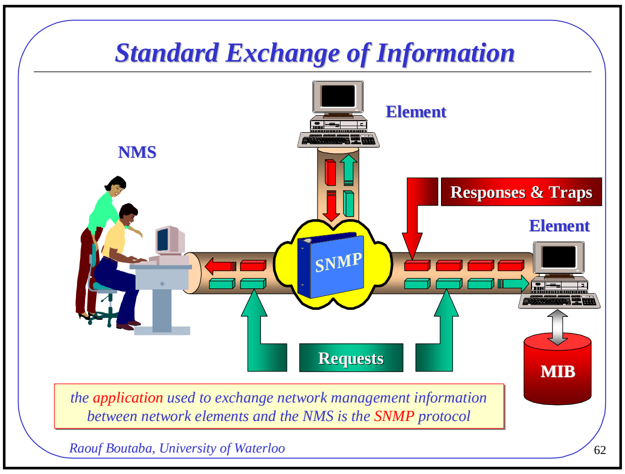


information is exchanged between elements of a network and the NMS



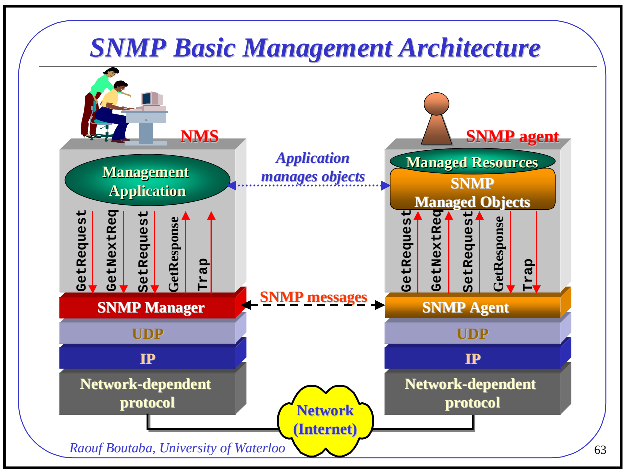
Exchange of Management Information

the network element accepts and processes any request from the NMS to access information held in the MIB

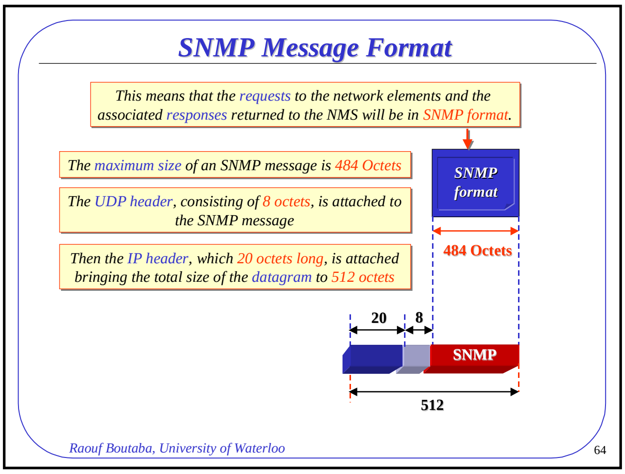


Standard Exchange of Information

the application used to exchange network management information between network elements and the NMS is the SNMP protocol



SNMP Basic Management Architecture



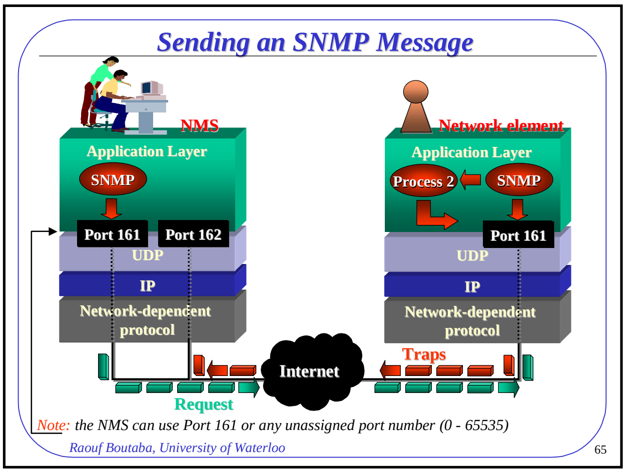
SNMP Message Format

This means that the requests to the network elements and the associated responses returned to the NMS will be in SNMP format.

The maximum size of an SNMP message is 484 Octets

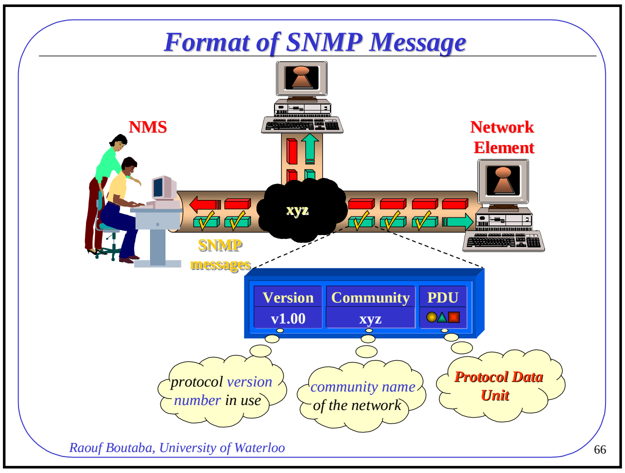
The UDP header, consisting of 8 octets, is attached to the SNMP message

Then the IP header, which 20 octets long, is attached bringing the total size of the datagram to 512 octets

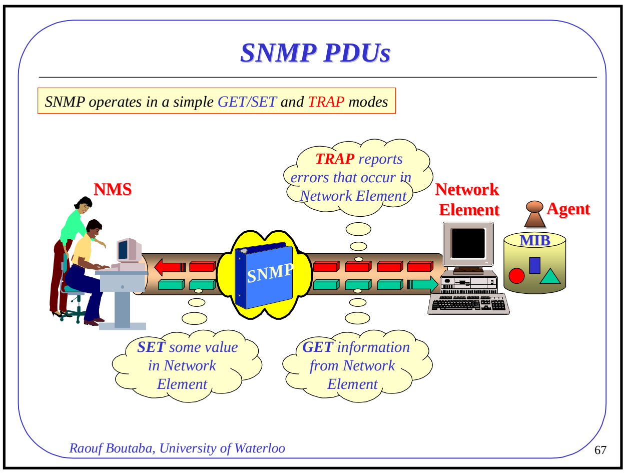


Sending an SNMP Message

Note: the NMS can use Port 161 or any unassigned port number (0 - 65535)



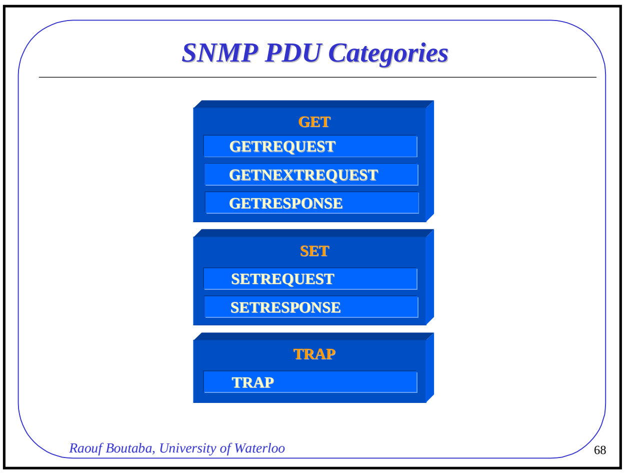
Format of SNMP Message



SNMP PDUs

SNMP operates in a simple GET/SET and TRAP modes

NMS
TRAP reports errors that occur in Network Element
Network Element
Agent
MIB
SNMP
SET some value in Network Element
GET information from Network Element

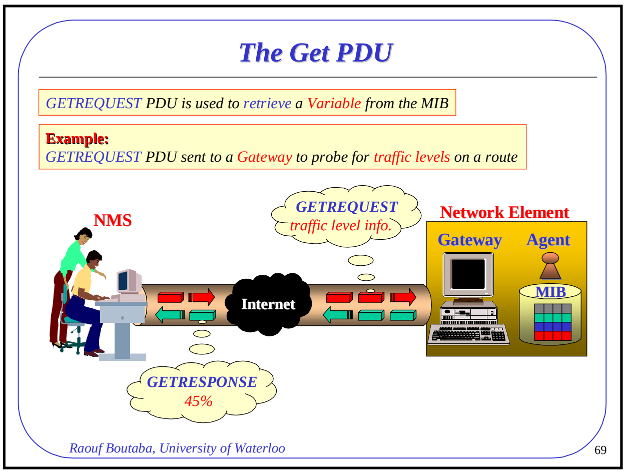


SNMP PDU Categories

GET
- GETREQUEST
- GETNEXTREQUEST
- GETRESPONSE

SET
- SETREQUEST
- SETRESPONSE

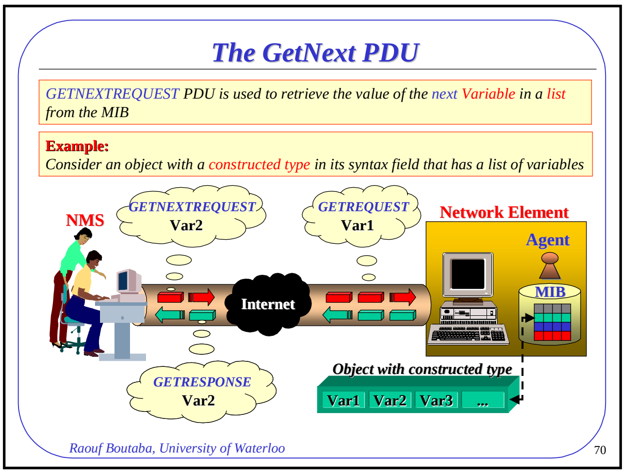
TRAP
- TRAP



The Get PDU

GETREQUEST PDU is used to retrieve a Variable from the MIB

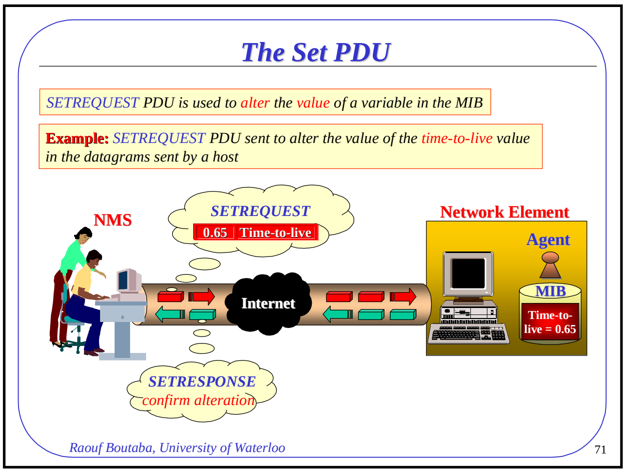
Example:
GETREQUEST PDU sent to a Gateway to probe for traffic levels on a route



The GetNext PDU

GETNEXTREQUEST PDU is used to retrieve the value of the next Variable in a list from the MIB

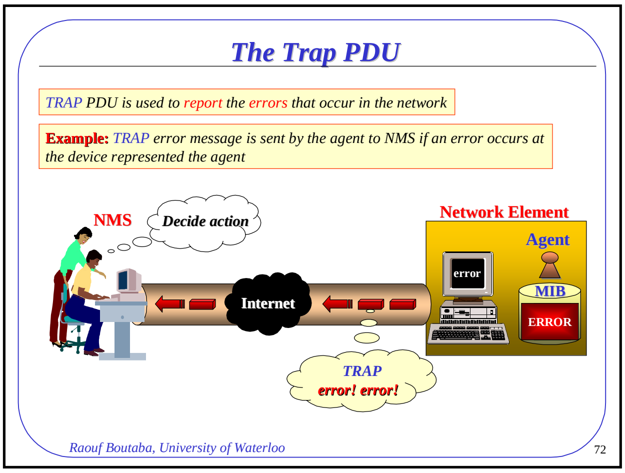
Example:
Consider an object with a constructed type in its syntax field that has a list of variables



The Set PDU

SETREQUEST PDU is used to alter the value of a variable in the MIB

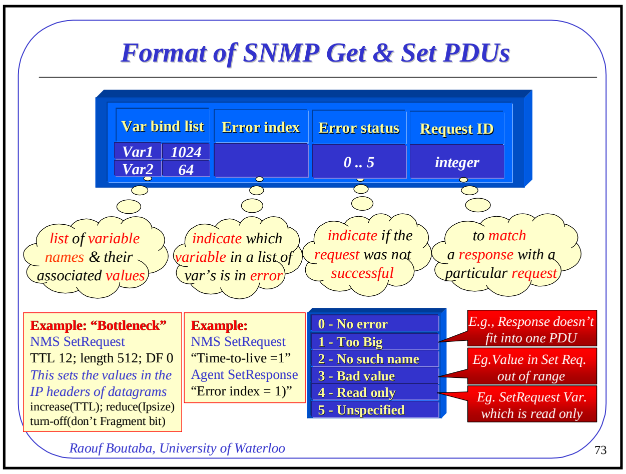
Example: *SETREQUEST PDU sent to alter the value of the time-to-live value in the datagrams sent by a host*



The Trap PDU

TRAP PDU is used to report the errors that occur in the network

Example: *TRAP error message is sent by the agent to NMS if an error occurs at the device represented the agent*



Format of SNMP Get & Set PDUs

Var bind list	Error index	Error status	Request ID
Var1 1024 Var2 64		0 .. 5	integer

list of variable names & their associated values
indicate which variable in a list of var's is in error
indicate if the request was not successful
to match a response with a particular request

Example: "Bottleneck"
NMS SetRequest
TTL 12; length 512; DF 0
This sets the values in the IP headers of datagrams
increase(TTL); reduce(Ipsize)
turn-off(don't Fragment bit)

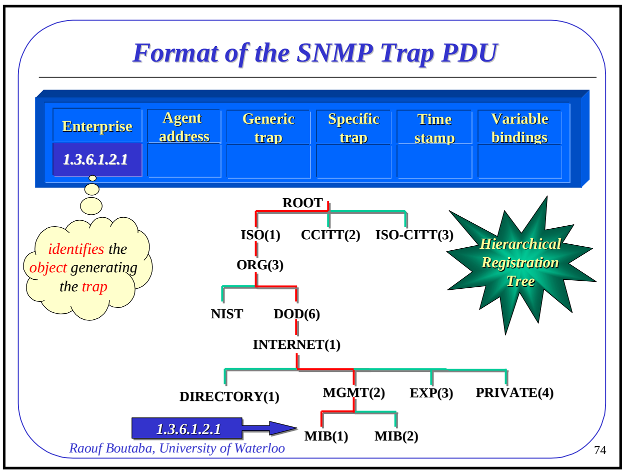
Example:
NMS SetRequest
"Time-to-live =1"
Agent SetResponse
"Error index = 1)"

- 0 - No error
- 1 - Too Big
- 2 - No such name
- 3 - Bad value
- 4 - Read only
- 5 - Unspecified

E.g., Response doesn't fit into one PDU

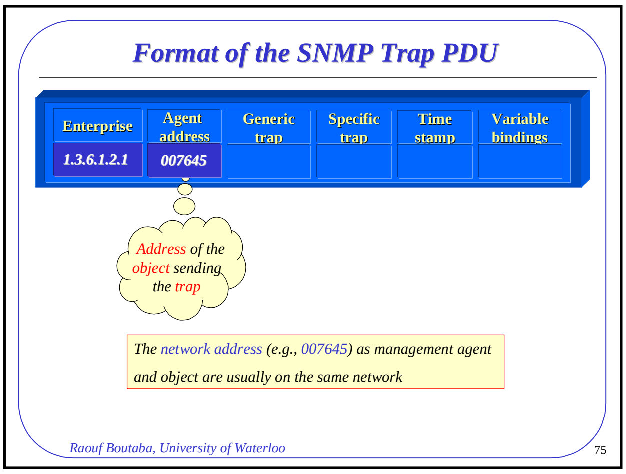
Eg.Value in Set Req. out of range

Eg. SetRequest Var. which is read only



Format of the SNMP Trap PDU

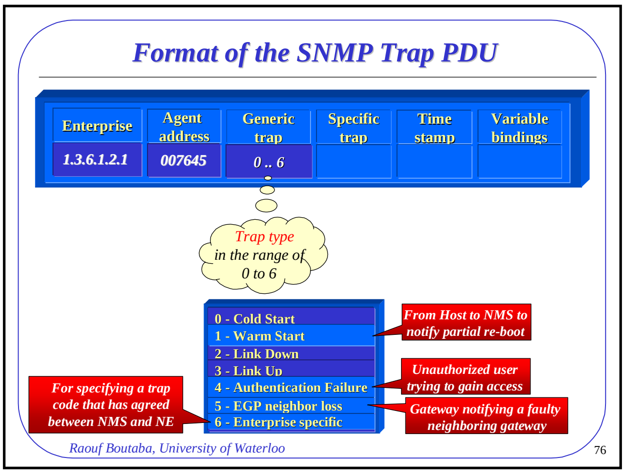
Enterprise	Agent address	Generic trap	Specific trap	Time stamp	Variable bindings
1.3.6.1.2.1					



Format of the SNMP Trap PDU

Enterprise	Agent address	Generic trap	Specific trap	Time stamp	Variable bindings
1.3.6.1.2.1	007645				

The network address (e.g., 007645) as management agent and object are usually on the same network



Format of the SNMP Trap PDU

Enterprise	Agent address	Generic trap	Specific trap	Time stamp	Variable bindings
1.3.6.1.2.1	007645	0 .. 6			

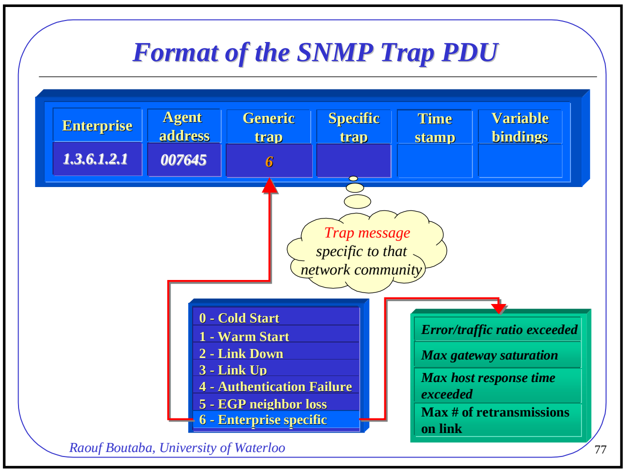
- 0 - Cold Start
- 1 - Warm Start
- 2 - Link Down
- 3 - Link Up
- 4 - Authentication Failure
- 5 - EGP neighbor loss
- 6 - Enterprise specific

From Host to NMS to notify partial re-boot

Unauthorized user trying to gain access

Gateway notifying a faulty neighboring gateway

For specifying a trap code that has agreed between NMS and NE

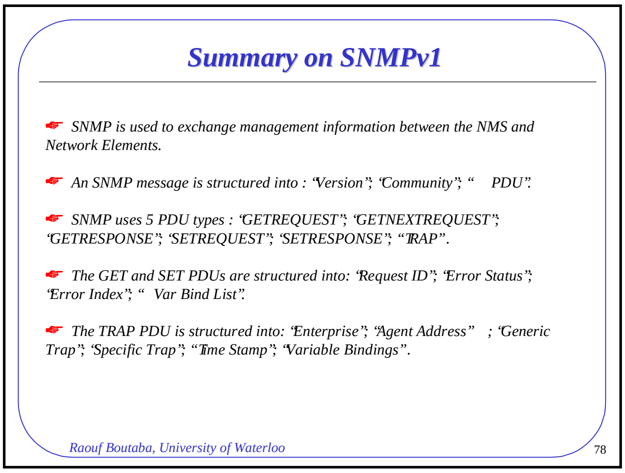


Format of the SNMP Trap PDU

Enterprise	Agent address	Generic trap	Specific trap	Time stamp	Variable bindings
1.3.6.1.2.1	007645	6			

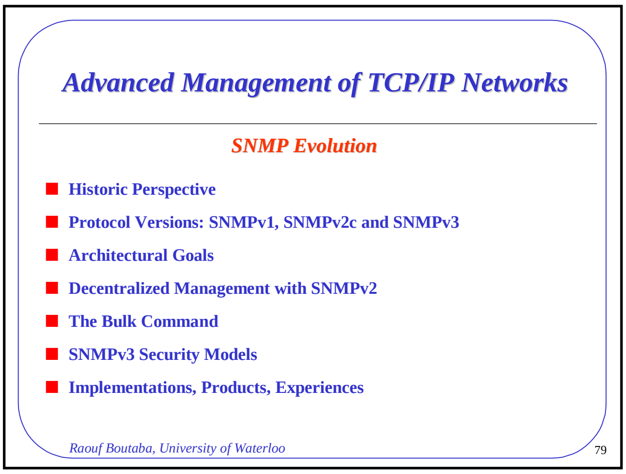
- 0 - Cold Start
- 1 - Warm Start
- 2 - Link Down
- 3 - Link Up
- 4 - Authentication Failure
- 5 - EGP neighbor loss
- 6 - Enterprise specific

- *Error/traffic ratio exceeded*
- *Max gateway saturation*
- *Max host response time exceeded*
- Max # of retransmissions on link



Summary on SNMPv1

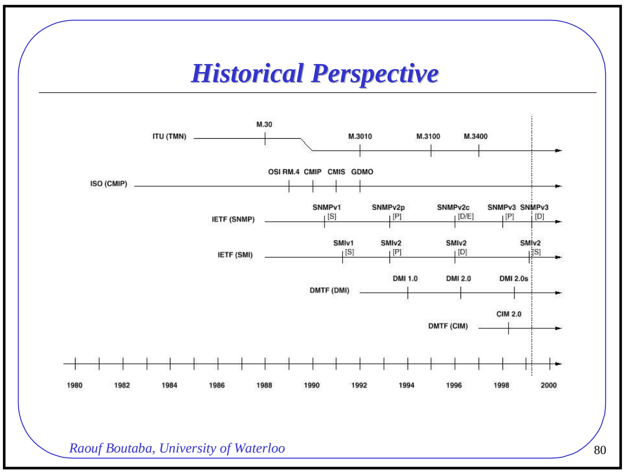
- *SNMP is used to exchange management information between the NMS and Network Elements.*
- *An SNMP message is structured into : "Version"; "Community"; "   PDU".*
- *SNMP uses 5 PDU types : "GETREQUEST"; "GETNEXTREQUEST"; "GETRESPONSE"; "SETREQUEST"; "SETRESPONSE"; "TRAP".*
- *The GET and SET PDUs are structured into: "Request ID"; "Error Status"; "Error Index"; " Var Bind List".*
- *The TRAP PDU is structured into: "Enterprise"; "Agent Address"   ; "Generic Trap"; "Specific Trap"; "Time Stamp"; "Variable Bindings".*



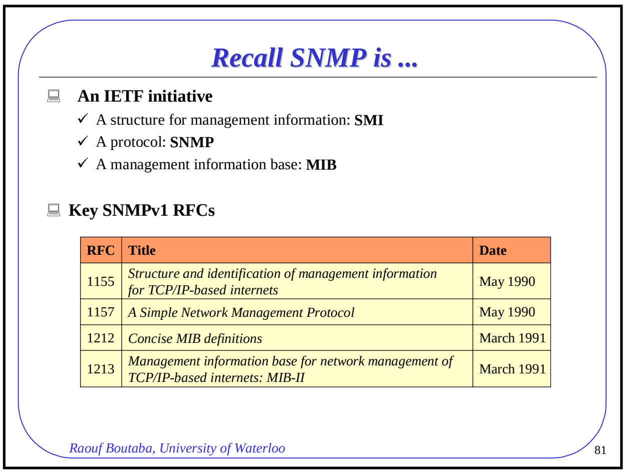
Advanced Management of TCP/IP Networks

SNMP Evolution

- Historic Perspective
- Protocol Versions: SNMPv1, SNMPv2c and SNMPv3
- Architectural Goals
- Decentralized Management with SNMPv2
- The Bulk Command
- SNMPv3 Security Models
- Implementations, Products, Experiences



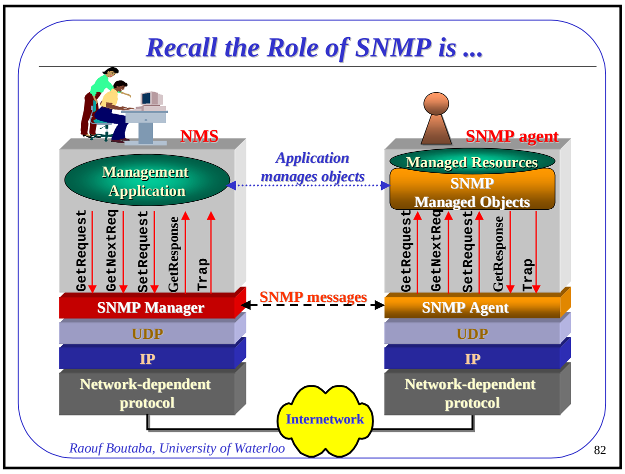
Historical Perspective



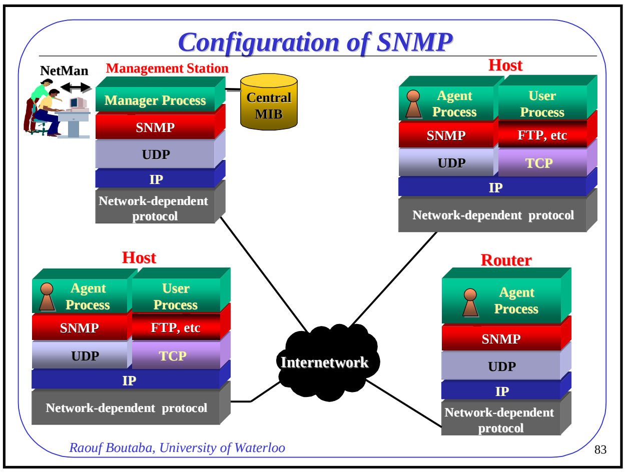
Recall SNMP is ...

- **An IETF initiative**
  - ✓ A structure for management information: **SMI**
  - ✓ A protocol: **SNMP**
  - ✓ A management information base: **MIB**
- **Key SNMPv1 RFCs**

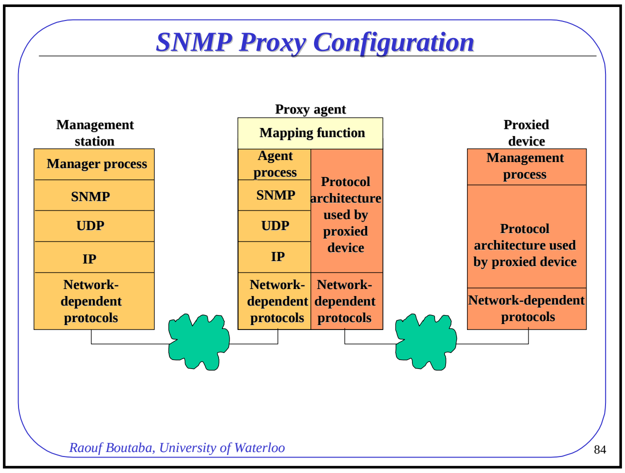
RFC	Title	Date
1155	*Structure and identification of management information for TCP/IP-based internets*	May 1990
1157	*A Simple Network Management Protocol*	May 1990
1212	*Concise MIB definitions*	March 1991
1213	*Management information base for network management of TCP/IP-based internets: MIB-II*	March 1991



Recall the Role of SNMP is ...

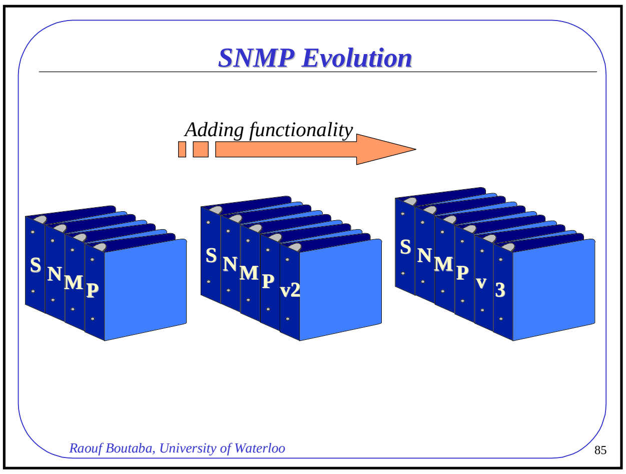


Configuration of SNMP



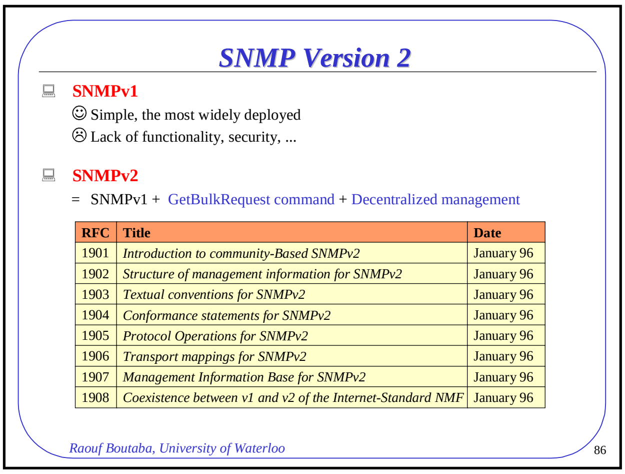
SNMP Proxy Configuration

Management station
Manager process
SNMP
UDP
IP
Network-dependent protocols
Proxy agent
Mapping function
Agent process
SNMP
UDP
IP
Network-dependent protocols
Protocol architecture used by proxied device
Network-dependent protocols
Proxied device
Management process
Protocol architecture used by proxied device
Network-dependent protocols



SNMP Evolution

Raouf Boutaba, University of Waterloo

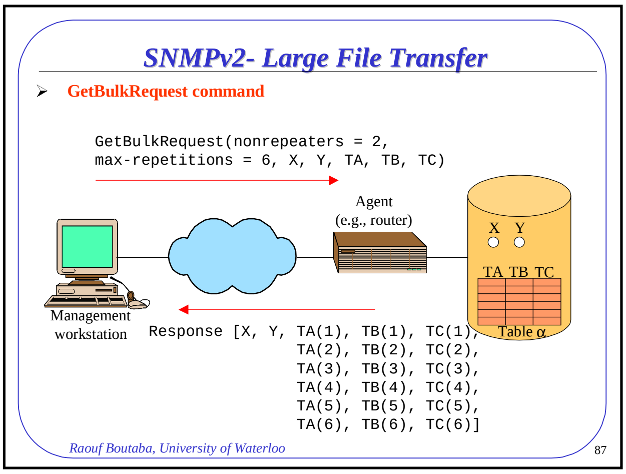


SNMP Version 2

- **SNMPv1**
  - ☺ Simple, the most widely deployed
  - ☹ Lack of functionality, security, ...
- **SNMPv2**
  - = SNMPv1 + GetBulkRequest command + Decentralized management

RFC	Title	Date
1901	*Introduction to community-Based SNMPv2*	January 96
1902	*Structure of management information for SNMPv2*	January 96
1903	*Textual conventions for SNMPv2*	January 96
1904	*Conformance statements for SNMPv2*	January 96
1905	*Protocol Operations for SNMPv2*	January 96
1906	*Transport mappings for SNMPv2*	January 96
1907	*Management Information Base for SNMPv2*	January 96
1908	*Coexistence between v1 and v2 of the Internet-Standard NMF*	January 96

Raouf Boutaba, University of Waterloo



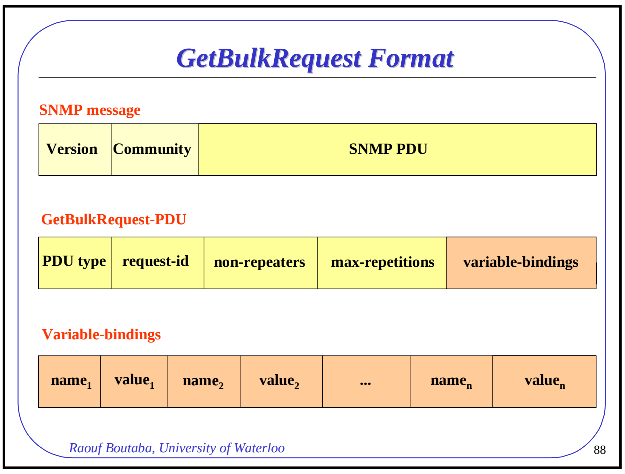
SNMPv2- Large File Transfer

- **GetBulkRequest command**

```
GetBulkRequest(nonrepeaters = 2,
max-repetitions = 6, X, Y, TA, TB, TC)
```

```
Response [X, Y, TA(1), TB(1), TC(1),
                TA(2), TB(2), TC(2),
                TA(3), TB(3), TC(3),
                TA(4), TB(4), TC(4),
                TA(5), TB(5), TC(5),
                TA(6), TB(6), TC(6)]
```

Raouf Boutaba, University of Waterloo



GetBulkRequest Format

SNMP message

Version	Community	SNMP PDU

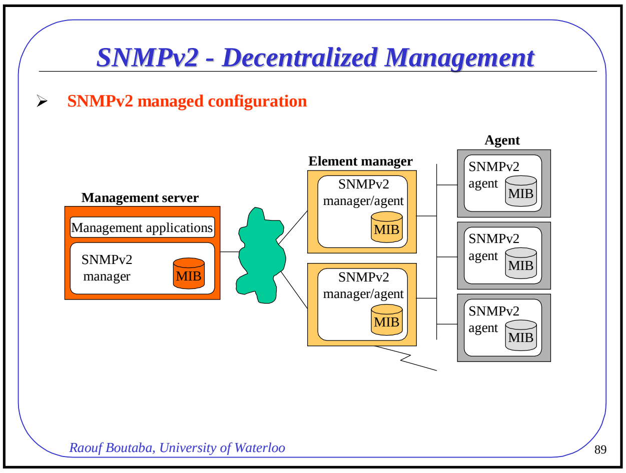
GetBulkRequest-PDU

PDU type	request-id	non-repeaters	max-repetitions	variable-bindings

Variable-bindings

name_1	value_1	name_2	value_2	...	name_n	value_n

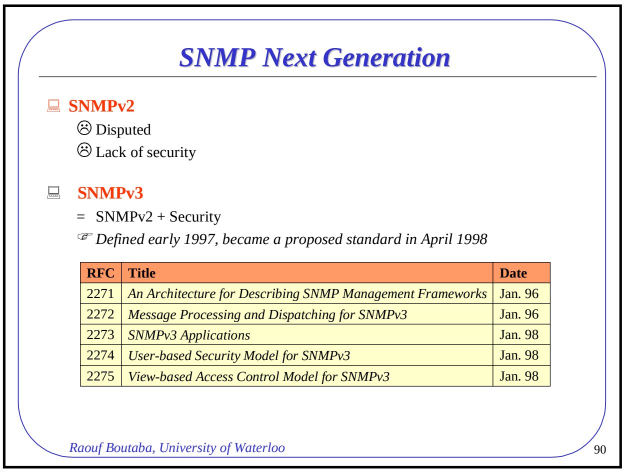
Raouf Boutaba, University of Waterloo



SNMPv2 - Decentralized Management

- **SNMPv2 managed configuration**

Raouf Boutaba, University of Waterloo



SNMP Next Generation

- **SNMPv2**
  - ☹ Disputed
  - ☹ Lack of security
- **SNMPv3**
  - = SNMPv2 + Security
  - ☞ *Defined early 1997, became a proposed standard in April 1998*

RFC	Title	Date
2271	*An Architecture for Describing SNMP Management Frameworks*	Jan. 96
2272	*Message Processing and Dispatching for SNMPv3*	Jan. 96
2273	*SNMPv3 Applications*	Jan. 98
2274	*User-based Security Model for SNMPv3*	Jan. 98
2275	*View-based Access Control Model for SNMPv3*	Jan. 98

Raouf Boutaba, University of Waterloo

SNMPv3 since 1998

- Data Definition Language
  - SMIv2 defined in RFC 2578-2580
- Definition of Management Information (MIBs)
  - Nearly 100 IETF MIB modules containing roughly 10.000 definitions
  - Even larger and growing number of enterprise-specific MIB modules
- Protocol Operations and Transport Mappings
  - RFC 1905-1907 (Draft Standard)
  - Currently under revision for full Internet Standard status
- Security and Administration
  - RFC 2271-2275 (Proposed Standard)
  - Publication of revised versions as Draft Standard in 1999

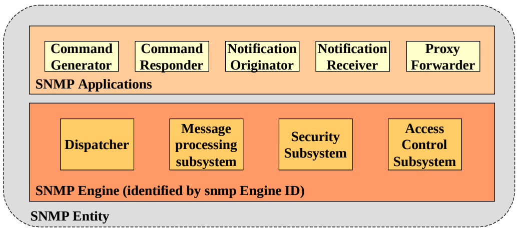
Architectural Goals of SNMPv3

- *Address the need for secure SNMP (write) operations*
- *Define an architecture that allows for longevity of SNMP frameworks*
- *Support inexpensive minimal conforming implementations*
- *Support more complex conforming implementations required in large networks*
- *Allow to move portions of the architecture along the IETF standards track*
- *Use existing materials as much as possible*
- *Keep SNMP as simple as possible*

SNMPv3 Framework

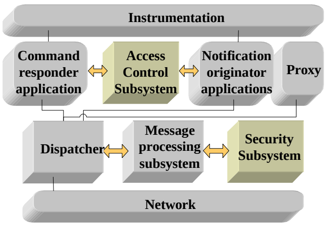
- Data Definition Language (SMI)
  - SMIv2 defined in RFC 2578-2580
- Definition of Management Information (MIBs)
  - Nearly 100 IETF MIB modules containing roughly 10.000 definitions
  - Even larger and growing number of enterprise-specific MIB modules
- Protocol Operations and Transport Mappings (SMI)
  - RFC 1905-1907 (Draft Standard)
  - under revision for full Internet Standard status
- Security and Administration
  - RFC 2271-2275 (Proposed Standard).

Architecture of SNMPv3 Entities

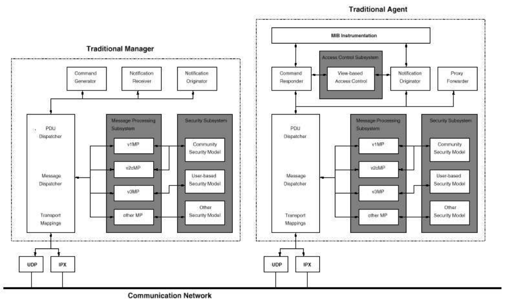


- ✓ Exactly one engine per SNMP entity and exactly one dispatcher per SNMP engine
- ✓ Every abstract subsystem may consist of one or more concrete models
- ✓ Modularization enables incremental enhancements to SNMP

Example: SNMPv3 Agent Entity

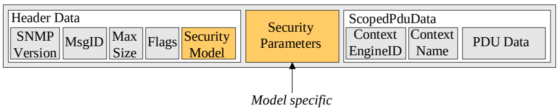

Instrumentation
Command responder application
Access Control Subsystem
Notification originator applications
Proxy
Dispatcher
Message processing subsystem
Security Subsystem
Network

Manager and Agent in the SNMP Architecture

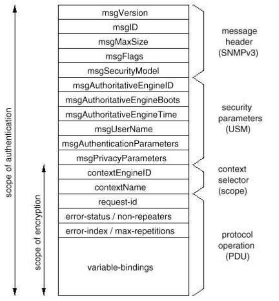


SNMPv3 Security

- **User Based Security Model (USM)**
  - Data integrity and Authentication
  - Privacy
- **View-based Access Control Model (VACM)**
  - Access control
- **SNMPv3 Message format**



SNMPv3/USM Messages (RFC 2272, 2274)

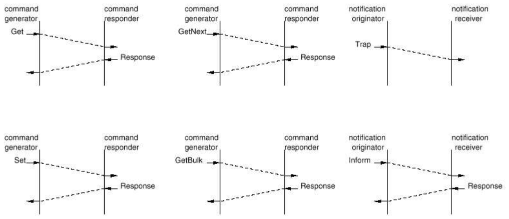


- `msgVersion` identifies the message processing model.
- `msgSecurityModel` identifies the security model.
- `contextEngineID` and `contextName` determine the context.
- The type of protocol operation (and version) is determined by the tag of the PDU.

SNMPv3 Contexts

- Context is a collection of management information accessible by an SNMP entity
  - An SNMP entity potentially has access to many contexts
  - An item of management information may exist in more than one context
- Within a management domain, a managed object is uniquely identified by:
  - the identification of the engine within the SNMP entity (e.g. 'x yz')
  - the context name within the SNMP entity (e.g. 'board1')
  - the managed object type (e.g. 'IF -MIB.ifDescr')
  - the instance identifier (e.g. '1')

SNMPv3 Protocol Operations (RFC 1905)



An additional Report protocol operation is used internally for error notifications, engine discovery and clock synchronization.

Classes of Protocol Operations

The processing of a message depends on the class of the embedded protocol operation:

Class	Description
Read	PDUs that retrieve management information.
Write	PDUs which attempt to modify management information.
Response	PDUs which are sent in response to a request.
Notification	PDUs which transmit event notifications.
Internal	PDUs exchanged internally between SNMP engines.
Confirmed	PDUs which cause the receiver to send a response.
Unconfirmed	PDUs which are not acknowledged.

- *The introduction of PDU classes enables the IETF to add new protocol operations without having to update the message processing specification.*
- *There is no explicit support in the message format to indicate the protocol operations supported/used by an SNMP engine.*

SNMPv3 Error and Exception Handling

SNMPv3 Exception	Get	GetNext/GetBulk	SNMPv1 Error Status
noSuchObject	X		noSuchName(2)
noSuchInstance	X		noSuchName(2)
endOfMibView		X	noSuchName(2)

Error handling in SNMPv1:

- An error response contains an error status and an error index.
- Error responses contain no useful management information.
- There is only a single error status and error index even if there are multiple errors.

Error and exception handling in SNMPv3:

- Per variable-binding exceptions in common error situations.
- One or more exceptions are not considered to be an error condition.
- A response with exceptions still contains useful management information.
- Other errors are handled as in SNMPv1 with more detailed error status codes.

An SNMPv3 command generator must be prepared to deal with SNMPv1 error responses that may come from proxied SNMPv1 command responders.

SNMPv3 and SNMPv1 Error Codes

SNMPv3 Error Code	Read Class	Write Class	Notification Class	SNMPv1 Error Code
noError(0)	X	X	X	noError(0)
tooBig(1)	X	X	X	tooBig(1)
noSuchName(2)				noSuchName(2)
badValue(3)				badValue(3)
readOnly(4)				readOnly(4)
genErr(5)	X	X	X	genErr(5)
noAccess(6)		X		noSuchName(2)
wrongType(7)		X		badValue(3)
wrongLength(8)		X		badValue(3)
wrongEncoding(9)		X		badValue(3)
wrongValue(10)		X		badValue(3)
noCreation(11)		X		noSuchName(2)
inconsistentValue(12)		X		badValue(3)
resourceUnavailable(13)		X		genErr(5)
commitFailed(14)		X		genErr(5)
undoFailed(15)		X		genErr(5)
authorizationError(16)	X	X	X	noSuchName(2)
notWritable(17)		X		noSuchName(2)
inconsistentName(18)		X		noSuchName(2)

SNMPv3/USM Textual Conventions

λ **SnmpEngineID**
- Unique identification of an SNMP engine within a management domain.

λ **SnmpSecurityModel**
- identification of a specific security model.

λ **SnmpMessageProcessingModel**
- Identification of a specific message processing model.
- The message processing model is encoded in the msgVersion.

λ **SnmpSecurityLevel**
- The security level of a given message (noAuthNoPriv, authNoPriv, authPriv).
- The security level is encoded in the msgFlags.

λ **KeyChange**
- Defines a cryptographic algorithm to change authentication or encryption keys.

Security Issues

v The following questions must be answered in order to decide whether an operation should be performed or not:

1. Is the message specifying an operation authentic?
2. Who requested the operation to be performed?
3. What objects are accessed in the operation?
4. What are the rights of the requester with regard to the objects of the operation?

v 1 and 2 are answered by message security mechanisms (authentication and privacy).

v 3 and 4 are answered by authorization mechanisms (access control).

USM Message Security (RFC 2274)

- Protection against the following threads:

  1. Modification of Information
     (Unauthorized modification of in-transit SNMP messages.)
  2. Masquerade
     (Unauthorized users attempting to use the identity of authorized users.)
  3. Disclosure
     (Protection against eavesdropping on the exchanges between SNMP entities.)
  4. Message Stream Modification
     (Re-ordered, delayed or replayed messages to affect unauthorized operations.)

- No protection against:
  - Denial of Service
    (Denial of service attacks are usually indistinguishable from network failures.)
  - Traffic Analysis
    (No significant advantage afforded by protecting against traffic analysis.)

USM Security Services (RFC 2274)

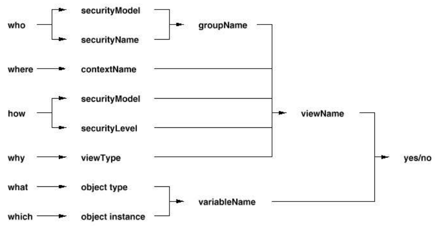
- **Data Integrity**
  - Data has not been altered or destroyed in an unauthorized manner.
  - Data sequences have not been altered to an extent greater than can occur non-maliciously.

- **Data Origin Authentication**
  - The claimed identity of the user on whose behalf received data was originated is corroborated.

- **Data Confidentiality**
  - Information is not made available or disclosed to unauthorized individuals, entities, or processes.

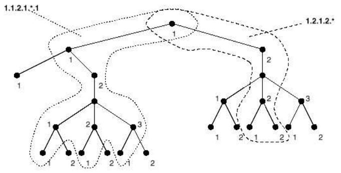
- **Message Timeliness and Limited Replay Protection**
  - A message whose generation time is outside of a time window is not accepted.
  - Message reordering is not dealt with and can occur in normal conditions too.

View-based Access Control Logic (RFC 2275)



- Three different securityLevels: noAuthNoPriv, authNoPriv, authPriv
- A securityName is a security model independent name for a principal.

View-based Access Control Views (RFC 2275)



- A view subtree is a set of managed object instances with a common OID prefix.
- A view tree family is the combination of an OID prefix with a bit mask.
- A bit of the bit mask defines whether an OID prefix component is significant or not (wild-carding).
- A view is an ordered set of view tree families.
- Access control rights are defined by a read view, write view or notify view.

Coexistence with SNMPv1 (and SNMPv2c)

- Community-based message processing model:
  - Integration of SNMPv1 (SNMPv2c) into the SNMP architecture.
  - Definition of a MIB for remote configuration of SNMPv1 (SNMPv2c) agents.
- Error code and exception mappings:
  - Mappings of SNMPv3 error codes and exceptions into SNMPv1 error codes.
- Handling of unsupported data types:
  - Unsupported data types are implicitly not in view.
- Conversion of trap messages:
  - All information contained in a SNMPv1 trap can be mapped into a SNMPv3 trap.
- SMI conversion from SMIv1 to SMIv2:
  - Guidelines for converting SMIv1 MIB modules into SMIv2 MIB modules.

Implementations, Products, Experiences

- Several implementations and products are available:

ACE*COMM	IBM
SNMP++v3 Project	InterWorking Labs
BMC Software	MG-SOFT Corporation
Cisco Systems	MultiPort Corporation
ISI/Epilogue	SNMP Research
Gambit Communications	TU Braunschweig
Halcyon	UC Davis
IBM Research	

- Experiences:
  - Configuring VACM manually is an error prone and time consuming task.
  - Remote configuration and key management requires not trivial applications.

Known Problems and Limitations of SNMPv3

- Missing extensibility for new base data types (e.g. Unsigned64).
- Missing extensibility for new protocol operations (e.g. GetSubtree).
- Limited flexibility for the definition of VACM rules.
- Asymmetries between notification filtering and VACM filtering.
- Positioning of security information in the middle of the message.
- Strength of USM security (DES versus Tripple-DES, key change procedure).
- Unnecessary complexity and misleading names in the message format definition.
- Insufficient performance gains compared to SNMPv1 (bulk data transfer).
- Degrees of freedom in complex write operations on tables are likely to cause interoperability problems.

Summary and Perspective

- Next Generation Structure of Management Information (SMIng)
- Bulk MIB Data Transfers
- Future of Internet Management
- References
- Links to Online Resources

Efficient Bulk Transfer of MIB-Data

Approach #1: SNMP extensions for bulk MIB data transfers

- Use TCP as a transport protocol.
- Compression of SNMP messages using gzip.
- Introduction of a new GetSubtree protocol operation.

Approach #2: SNMP in conjunction with FTP

- Definition of a MIB for storing MIB data in local files.
- Definition of a MIB for initiating FTP transfers.

Approach #3: Alternate protocols

- Definition of a MIME type for carrying MIB data.
- Transfer of MIME encapsulated MIB data via HTTP or SMTP.

Future of Internet Management

Things that may be useful (short term):

- Standardized APIs for SNMP and for accessing MIB definitions?
- Protocols and APIs for exchanging topology and configuration information?
- Protocols and APIs for exchanging alarm and trouble ticket records?
- SNMP version 4 (really?)
- Alternate protocols to exchange management information?

Longer term perspective:

- Less is more ==> Self-managing devices and networks?
- What are the alternatives? CORBA? CIM? CMIP/GDMO/TMN?
- What about active networks and intelligent mobile agents?

Request for Comments (RFCs)

D. Harrington, R. Presuhn, B. Wijnen: An Architecture for Describing SNMP Management Frameworks, RFC 2271, January 1998

J. Case, D. Harrington, R. Presuhn, and B. Wijnen, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP), RFC 2272, January 1998

D. Levi, P. Meyer, B. Stewart: SNMPv3 Applications, RFC 2273, January 1998

U. Blumenthal, B. Wijnen: User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3), RFC 2274, January 1998.

B. Wijnen, R. Presuhn, K. McCloghrie: View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP), RFC 2275, January 1998.

J. Case, K. McCloghrie, M. Rose, S. Waldbusser, Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2), RFC 1905, January 1996

J. Case, K. McCloghrie, M. Rose, S. Waldbusser, Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2), RFC 1906, January 1996

J. Case, K. McCloghrie, M. Rose, S. Waldbusser, Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2), RFC 1907, January 1996

J. Case, R. Mundy, D. Partain, B. Stewart: Introduction to Version 3 of the Internet-standard Network Management Framework, RFC (to be published), 1999

R. Frye, D. Levi, S. Routhier, B. Wijnen: Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework, RFC (to be published), 1999

Request for Comments (RFCs)

K. McCloghrie, D. Perkins, J. Sch• onw• alder, J. Case, M. Rose, S. Waldbusser: Structure of Management Information Version 2 (SMIv2), STD 58, RFC 2578, April 1999

K. McCloghrie, D. Perkins, J. Sch• onw• alder, J. Case, M. Rose, S. Waldbusser: Textual Conventions for SMIv2, STD 58, RFC 2579, April 1999

K. McCloghrie, D. Perkins, J. Sch• onw• alder, J. Case, M. Rose, S. Waldbusser: Conformance Statements for SMIv2, STD 58, RFC 2580, April 1999

M. Daniele, B. Wijnen, and D. Francisco: Agent Extensibility (AgentX) Protocol Version 1, RFC 2257, January 1998

D.B. Levi and J. Sch• onw• alder: Definitions of Managed Objects for the Delegation of Management Scripts, RFC (to be published), 1999

D.B. Levi and J. Sch• onw• alder: Definitions of Managed Objects for Scheduling Management Operations, RFC (to be published), 1999

J. Sch• onw• alder, J. Quittek: Script MIB Extensibility Protocol Version 1.0, RFC (to be published), 1999

B. Stewart, Expression MIB, (work in progress), 1999

B. Stewart, Event MIB, (work in progress), 1999

B. Stewart, Notification Log MIB, (work in progress), 1999

K. White, Definitions of Managed Objects for Remote Ping, Traceroute, and Name Lookup Operations, (work in progress), 1999

Books and Articles

W. Stallings: SNMP, SNMPv2, SNMPv3, and RMON 1 and 2, Addison-Wesley, 1999

D. Zeltserman: A Practical Guide to SNMPv3 and Network Management, Prentice Hall, 1999

D. Perkins and E. McGinnis: Understanding SNMP MIBs, Prentice Hall, 1997

The SimpleTimes, Special Issue on Agent Extensibility, SimpleTimes 4(2), April 1996

The SimpleTimes, Special Issue on SNMP Version 3, SimpleTimes 5(1), December 1997

M. White, S. Gudur: An Overview of the AgentX Protocol, SimpleTimes 6(1), April 1998

U. Blumenthal, N.C. Hien, B. Wijnen: Key derivation for network management applications, IEEE Network Magazine, 11(4), 1997

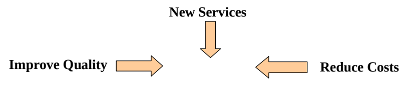
Management of Telecom Networks

- **ISO/OSI Network Management**
- **ITU-T/TMN, the Telecommunications Management Network**
- **Network Management Fora & Consortia (OSI/NM-F, TINA-C, OMG TSI, TMF)**

Telecommunications Market: What are the pressures ?

– Rapid technological and regulatory changes
…New risks, new costs, new competition

+ An expanding market
…Arrival of capacity greedy services (e.g., WWW, multimedi a services)



① Provide high quality services
② Control operating costs
➥ *Efficient management of telecommunication network and services*

ISO/OSI Network Management Standards

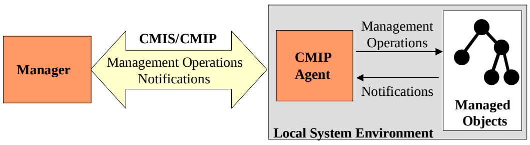
- *OSI Management Framework*
- *OSI Managed Object Model*
- *OSI Information Modeling*
- *OSI Communications Model*
- *Example Configuring a Circuit*

ISO/OSI Management Framework

ISO Management Standard

✓ *Initially to manage OSI protocols*
✓ *Known as the X700 series jointly developed with ITU SG 7 (late 80's/early 90's)*
✓ *Defines Functional ('FCAPS'), Information and Communication com ponents*

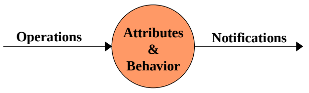
OSI systems management overview



OSI Managed Object Model

➢ **Information Modeling:**

Resource to be managed = **Managed Object (MO)**



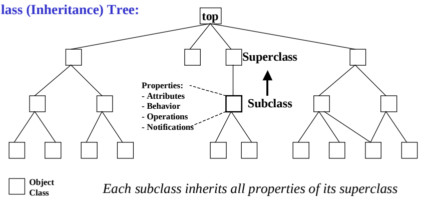
- **Attributes** *describe managed object state*
- **Management Operations** *which may be applied to object*
- **Behavior** *exhibited by object*
- **Notifications** *emitted by object*

A managed object **class** definition specifies these four properties. **Instances** of a managed object class share the same properties

OSI Information Modeling

- **GDMO** - ***G****uidelines for the* ***D****efinition of* ***M****anaged* ***O****bjects (MOs description)*
- **MIB** - ***M****anagement* ***I****nformation* ***B****ase (MOs store)*
- **MIT** - ***M****anagement* ***I****nformation* ***T****ree (Naming hierarchy)*
- **GRM** - ***G****eneral* ***R****elationship* ***M****odel (between MOs)*

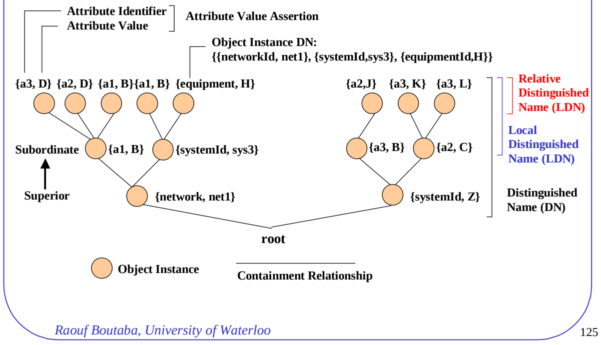
Object Class (Inheritance) Tree:



Each subclass inherits all properties of its superclass

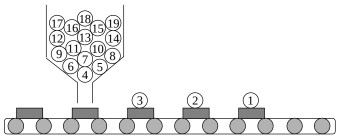
OSI Information Modeling (Cont'd)

Naming Tree (also known as Containment Tree and MIT):



OSI Information Modeling (Cont'd)

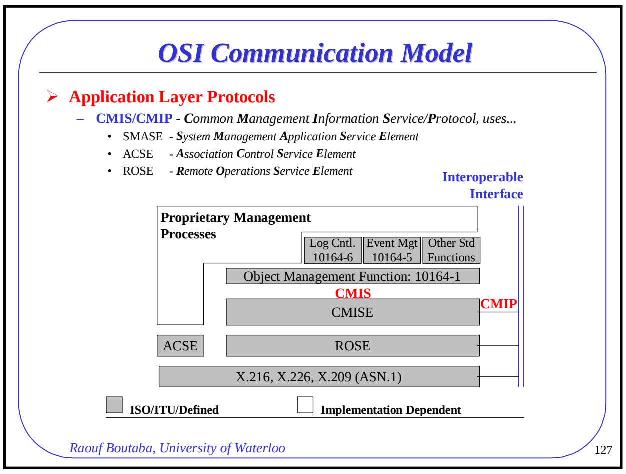
Object Registration



Registration assigns globally unique identifiers to items

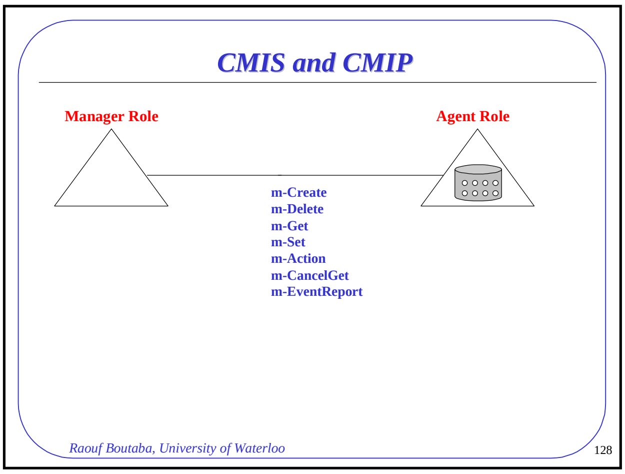
Once registered, an item cannot be changed

Many organizations around the world are authorized to act as registration authorities

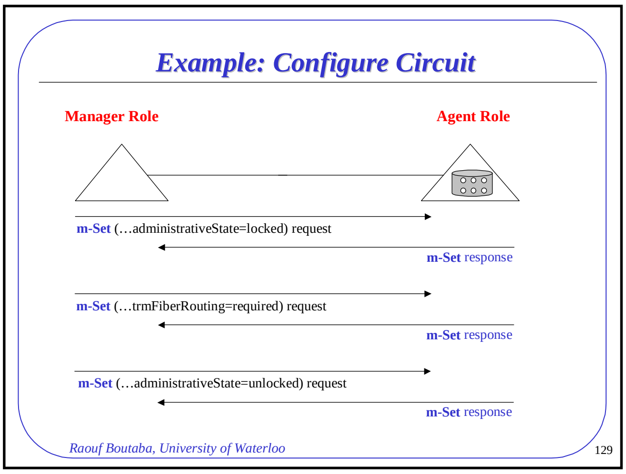


OSI Communication Model

- Application Layer Protocols
  - CMIS/CMIP - *Common Management Information Service/Protocol, uses...*
    - SMASE - *System Management Application Service Element*
    - ACSE - *Association Control Service Element*
    - ROSE - *Remote Operations Service Element*

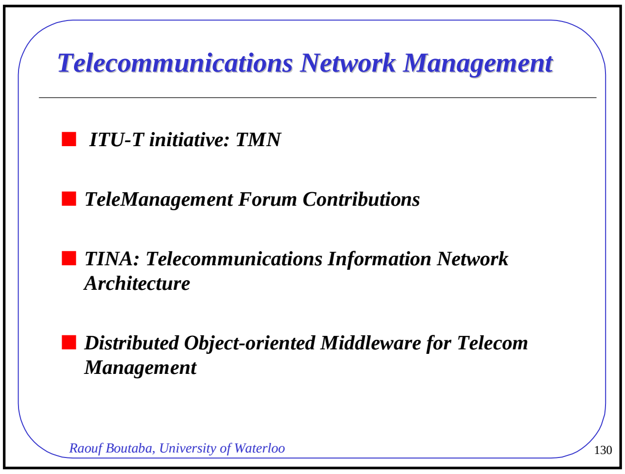


CMIS and CMIP



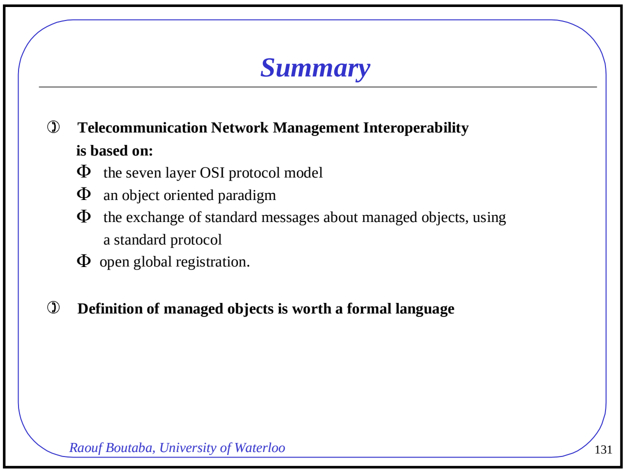
Example: Configure Circuit

Manager Role
Agent Role
m-Set (…administrativeState=locked) request
m-Set response
m-Set (…trmFiberRouting=required) request
m-Set response
m-Set (…administrativeState=unlocked) request
m-Set response



Telecommunications Network Management

- *ITU-T initiative: TMN*
- *TeleManagement Forum Contributions*
- *TINA: Telecommunications Information Network Architecture*
- *Distributed Object-oriented Middleware for Telecom Management*



Summary

➂ **Telecommunication Network Management Interoperability is based on:**

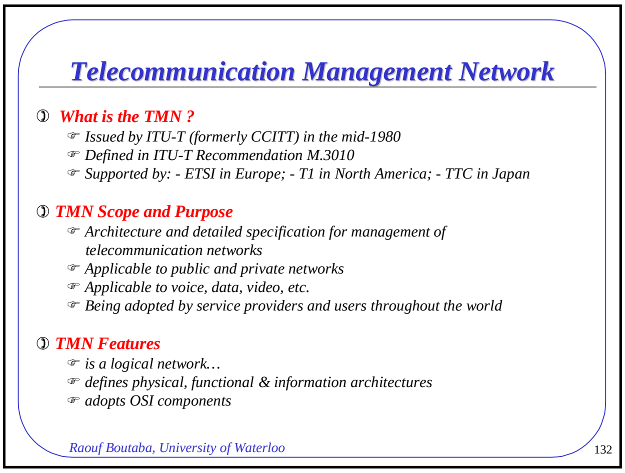
Φ the seven layer OSI protocol model

Φ an object oriented paradigm

Φ the exchange of standard messages about managed objects, using a standard protocol

Φ open global registration.

➂ **Definition of managed objects is worth a formal language**



Telecommunication Management Network

➂ *What is the TMN ?*

☞ *Issued by ITU-T (formerly CCITT) in the mid-1980*

☞ *Defined in ITU-T Recommendation M.3010*

☞ *Supported by: - ETSI in Europe; - T1 in North America; - TTC in Japan*

➂ *TMN Scope and Purpose*

☞ *Architecture and detailed specification for management of telecommunication networks*

☞ *Applicable to public and private networks*

☞ *Applicable to voice, data, video, etc.*

☞ *Being adopted by service providers and users throughout the world*

➂ *TMN Features*

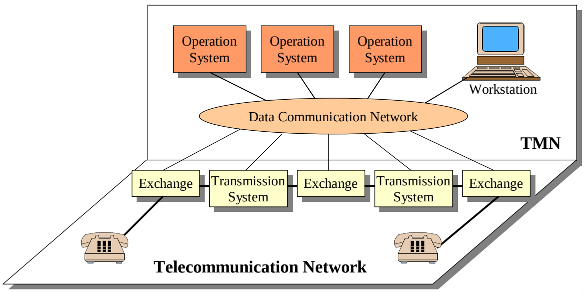
☞ *is a logical network…*

☞ *defines physical, functional & information architectures*

☞ *adopts OSI components*

Relationship of TMN to Telecommunication Network

- **TMN : A logical network ...**



Distinction between TMN and Managed Network

- **TMN**
  - → TMN is modeled as a network distinct from network being managed
  - → ‘Out of Band” Management
  - → TMN may use services or elements of managed network
  - → Overload or failure of managed network does not necessarily affect TMN
- **Telecommunications Network**
  - → Exchanges (switches)
  - → Transmission systems
  - → Terminal equipment
  - → Signaling systems
  - → Area Networks (LANs, MANs, WANs)
  - → Environmental (e.g. fans, power, air conditioning, etc.)
  - → Services and Applications
  - → TMN

TMN Management Functions

- **Performance Management**
- **Fault (or Maintenance) Management**
- **Configuration Management**
- **Accounting Management**
- **Security Management**

Performance Management

‘evaluate and report upon the behavior of telecommunications equipment and the effectiveness of the network or network element”

- **Performance Monitoring**
- **Performance Analysis**
- **Performance Management Control**

Fault (or Maintenance) Management

‘enable the detection, isolation, and correction of abnormal ope rations”

- **Alarm Surveillance**
- **Fault Localization**
- **Fault Correction**
- **Testing**
- **Trouble Administration**

Configuration Management

‘Exercise control over, identify, collect data from and provide data to network elements”

- **Installation**
- **Provisioning**
- **Changes**

Accounting Management

“enable the use of the network service to be measured and the costs for such use to be determined”

- **Billing Functions**
- **Tariffing Functions**

Security Management

“detect and prevent access to network and network management resources by unauthorized users”

- **Access security**
- **Security alarms**
- **Intrusion recovery**

TMN Logical Layered Architecture

Business Management

- *Enterprise View*
- *Goal Setting, finance, budgeting*
- *Product & human resource planning*

Service Management

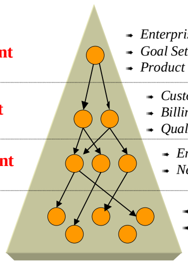
- *Customer contact/support*
- *Billing*
- *Quality of Service*

Network Management

- *End-to-end network view/management.*
- *Network support of Services*

Network Element Management

- *Network element view/management*
- *Adaptation/Mediation*



Network Element Layer

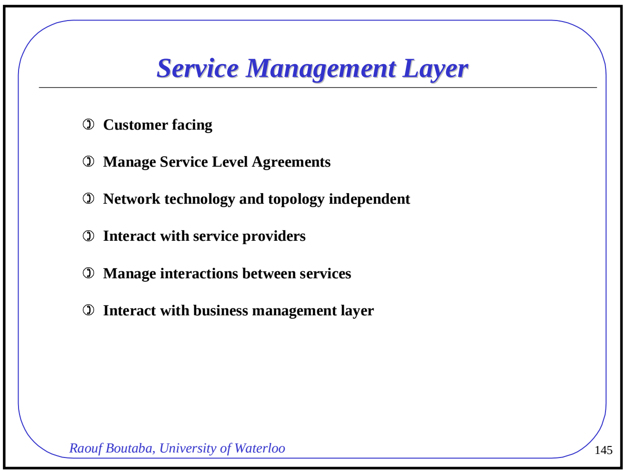
- **Provides actual network functions**
- **Mix of standard and proprietary features**
- **Examples:**
  - → **switch**
  - → **signaling transfer point**
  - → **multiplexer**
  - → **computer**

Network Element Management Layer

- **Manages subset of network element**
- **Usually technology specific (e.g. SONET multiplexer)**
- **Frequently vendor specific**
- **May provide consistency across different models or versions of network elements**
- **Gateway to network management layer**

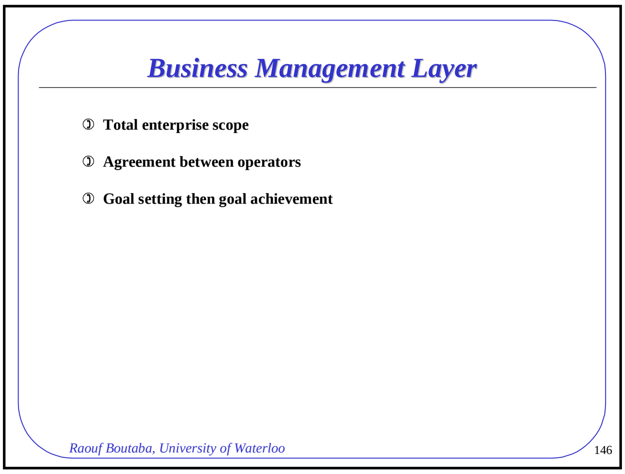
Network Management Layer

- **Management of network view of many network elements**
  - → **multiple technologies**
  - → **multiple vendors**
- **Manage network capabilities to provide services to customers**
- **Interact with service management layer**



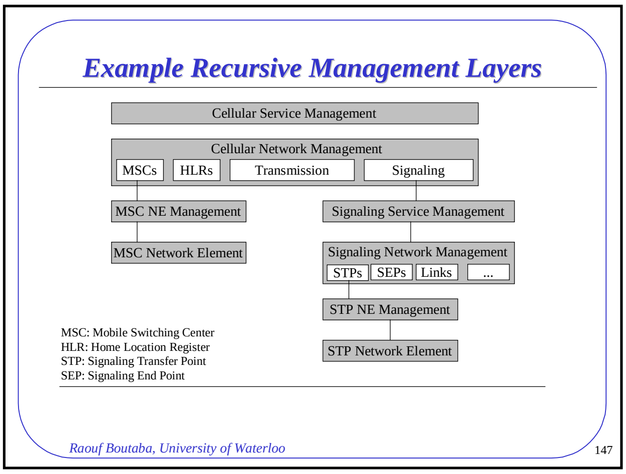
Service Management Layer

- Customer facing
- Manage Service Level Agreements
- Network technology and topology independent
- Interact with service providers
- Manage interactions between services
- Interact with business management layer



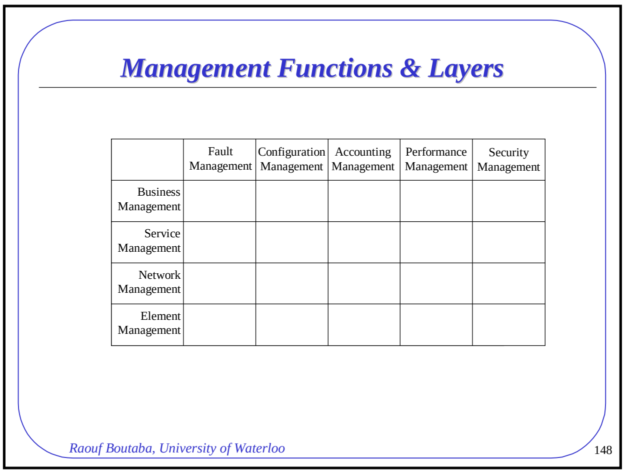
Business Management Layer

- Total enterprise scope
- Agreement between operators
- Goal setting then goal achievement



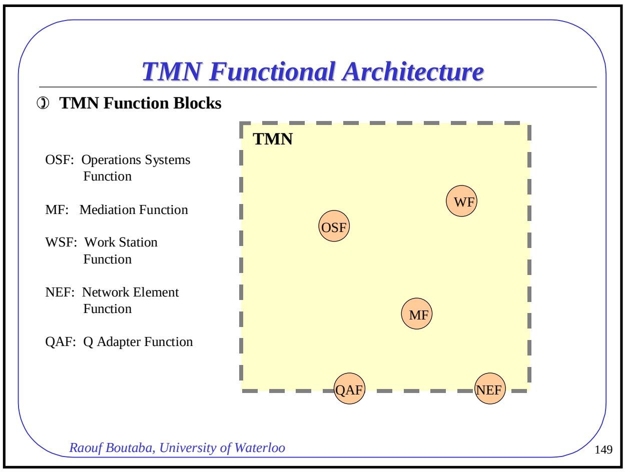
Example Recursive Management Layers

MSC: Mobile Switching Center
HLR: Home Location Register
STP: Signaling Transfer Point
SEP: Signaling End Point



Management Functions & Layers

	Fault Management	Configuration Management	Accounting Management	Performance Management	Security Management
Business Management					
Service Management					
Network Management					
Element Management					



TMN Functional Architecture

- TMN Function Blocks

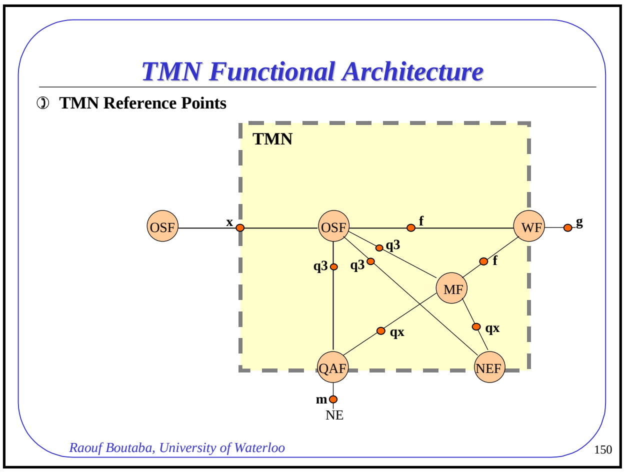
OSF: Operations Systems Function

MF: Mediation Function

WSF: Work Station Function

NEF: Network Element Function

QAF: Q Adapter Function

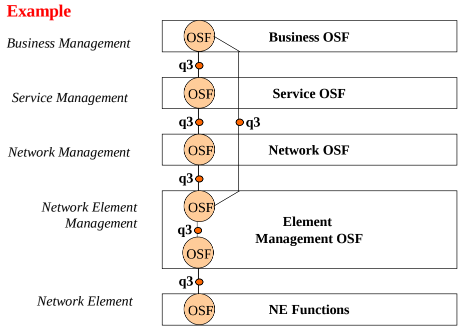


TMN Functional Architecture

- TMN Reference Points

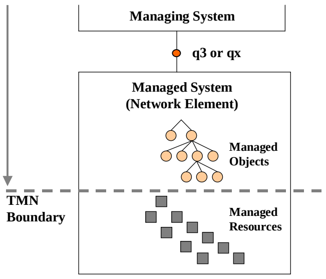
TMN Functional Hierarchy

- **Example**

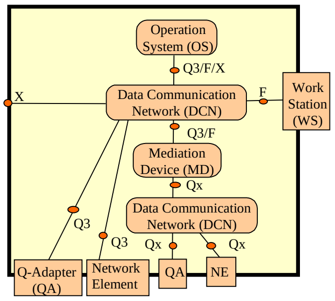


TMN Functional Architecture

- Network Element, Managed Objects, and Managed Object Resources



TMN Physical Architecture



TMN Q3 Interface

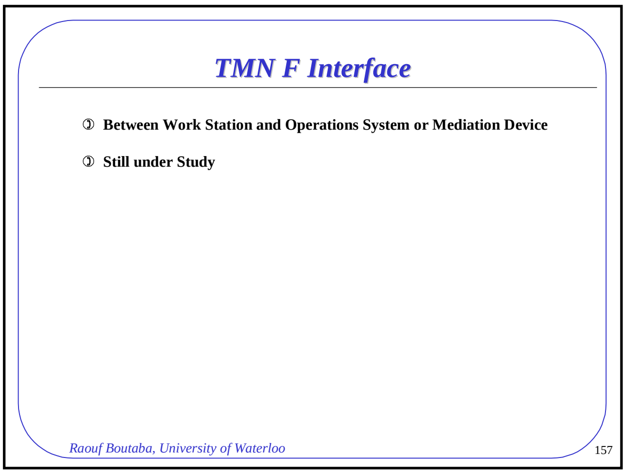
- Between Operations System and Mediation Device, Q Adapter, or Network Element
- X.700: CMIS/CMIP, GDMO Objects, etc.
- Managed Object Classes dependent on Network
- Common information model across multiple network elements

TMN Qx Interface

- Between Mediation Device and Q Adapter or Network Element
- Very similar to Q3
- X.700: CMIS/CMIP, GDMO Objects, etc.
- Lower Layer Protocols require mediation device
- May have simpler information model than Q3

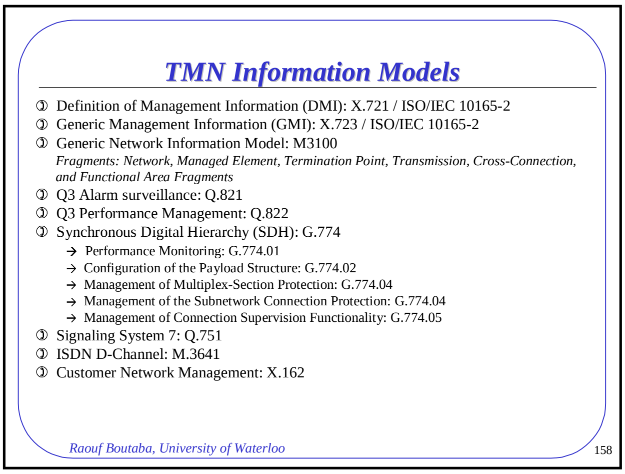
TMN X Interface

- Between two TMNs, e.g.:
  - → Distinct management domains
  - → Service provider to service provider
- X.700: CMIS/CMIP, GDMO Objects, etc.
- More extensive security requirements



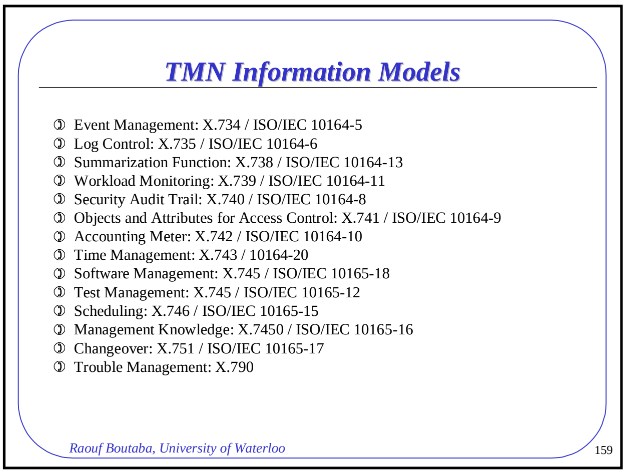
TMN F Interface

- Between Work Station and Operations System or Mediation Device
- Still under Study



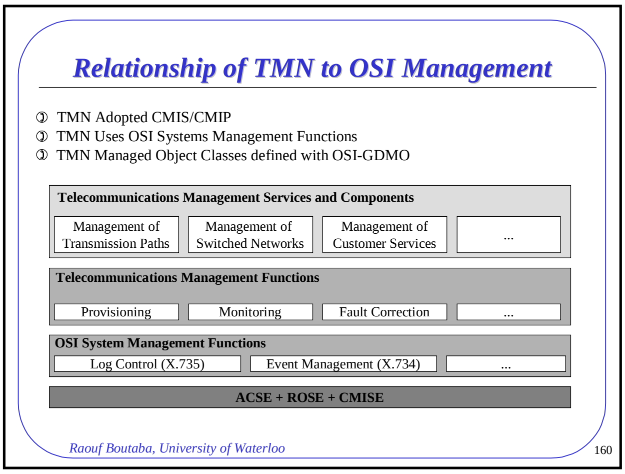
TMN Information Models

- Definition of Management Information (DMI): X.721 / ISO/IEC 10165-2
- Generic Management Information (GMI): X.723 / ISO/IEC 10165-2
- Generic Network Information Model: M3100
  Fragments: Network, Managed Element, Termination Point, Transmission, Cross-Connection, and Functional Area Fragments
- Q3 Alarm surveillance: Q.821
- Q3 Performance Management: Q.822
- Synchronous Digital Hierarchy (SDH): G.774
  - → Performance Monitoring: G.774.01
  - → Configuration of the Payload Structure: G.774.02
  - → Management of Multiplex-Section Protection: G.774.04
  - → Management of the Subnetwork Connection Protection: G.774.04
  - → Management of Connection Supervision Functionality: G.774.05
- Signaling System 7: Q.751
- ISDN D-Channel: M.3641
- Customer Network Management: X.162



TMN Information Models

- Event Management: X.734 / ISO/IEC 10164-5
- Log Control: X.735 / ISO/IEC 10164-6
- Summarization Function: X.738 / ISO/IEC 10164-13
- Workload Monitoring: X.739 / ISO/IEC 10164-11
- Security Audit Trail: X.740 / ISO/IEC 10164-8
- Objects and Attributes for Access Control: X.741 / ISO/IEC 10164-9
- Accounting Meter: X.742 / ISO/IEC 10164-10
- Time Management: X.743 / 10164-20
- Software Management: X.745 / ISO/IEC 10165-18
- Test Management: X.745 / ISO/IEC 10165-12
- Scheduling: X.746 / ISO/IEC 10165-15
- Management Knowledge: X.7450 / ISO/IEC 10165-16
- Changeover: X.751 / ISO/IEC 10165-17
- Trouble Management: X.790



Relationship of TMN to OSI Management

- TMN Adopted CMIS/CMIP
- TMN Uses OSI Systems Management Functions
- TMN Managed Object Classes defined with OSI-GDMO

Telecommunications Management Services and Components

Management of Transmission Paths | Management of Switched Networks | Management of Customer Services | ...

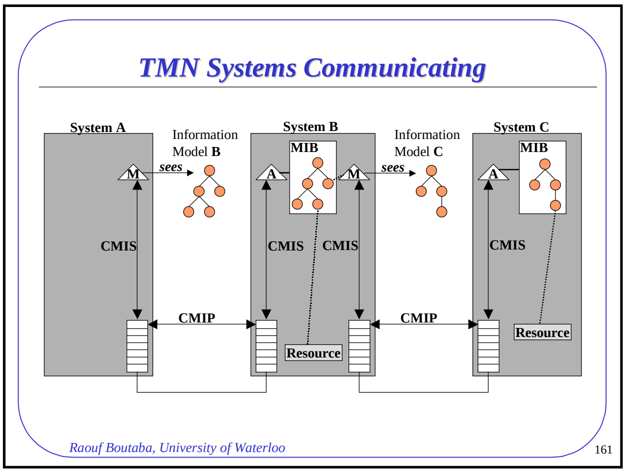
Telecommunications Management Functions

Provisioning | Monitoring | Fault Correction | ...

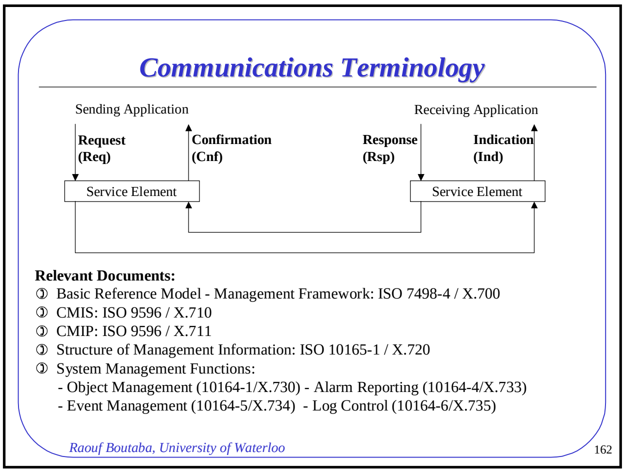
OSI System Management Functions

Log Control (X.735) | Event Management (X.734) | ...

ACSE + ROSE + CMISE



TMN Systems Communicating



Communications Terminology

Sending Application

Receiving Application

Request (Req)

Confirmation (Cnf)

Response (Rsp)

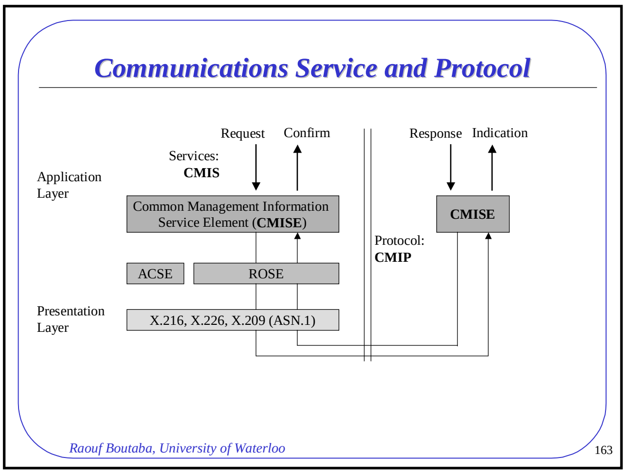
Indication (Ind)

Service Element

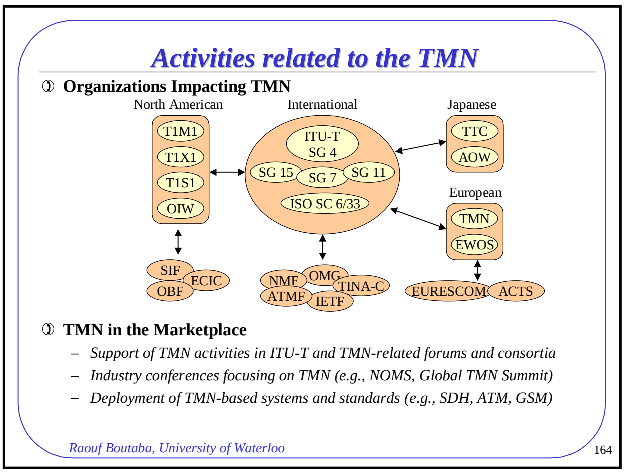
Service Element

Relevant Documents:

- Basic Reference Model - Management Framework: ISO 7498-4 / X.700
- CMIS: ISO 9596 / X.710
- CMIP: ISO 9596 / X.711
- Structure of Management Information: ISO 10165-1 / X.720
- System Management Functions:
  - Object Management (10164-1/X.730) - Alarm Reporting (10164-4/X.733)
  - Event Management (10164-5/X.734) - Log Control (10164-6/X.735)



Communications Service and Protocol

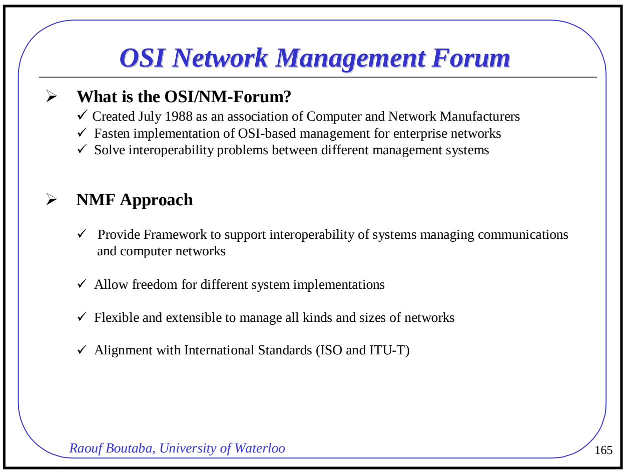


Activities related to the TMN

➂ **Organizations Impacting TMN**

➂ **TMN in the Marketplace**

- *Support of TMN activities in ITU-T and TMN-related forums and consortia*
- *Industry conferences focusing on TMN (e.g., NOMS, Global TMN Summit)*
- *Deployment of TMN-based systems and standards (e.g., SDH, ATM, GSM)*



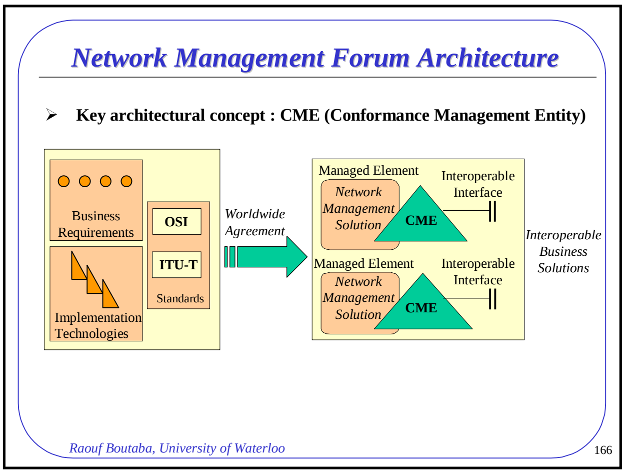
OSI Network Management Forum

➢ **What is the OSI/NM-Forum?**

- ✓ Created July 1988 as an association of Computer and Network Manufacturers
- ✓ Fasten implementation of OSI-based management for enterprise networks
- ✓ Solve interoperability problems between different management systems

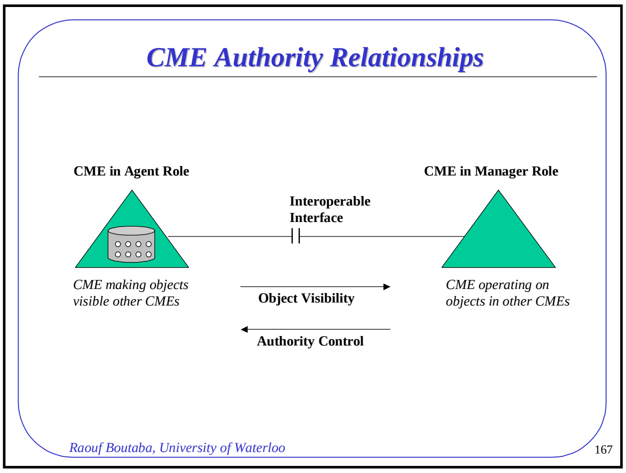
➢ **NMF Approach**

- ✓ Provide Framework to support interoperability of systems managing communications and computer networks
- ✓ Allow freedom for different system implementations
- ✓ Flexible and extensible to manage all kinds and sizes of networks
- ✓ Alignment with International Standards (ISO and ITU-T)

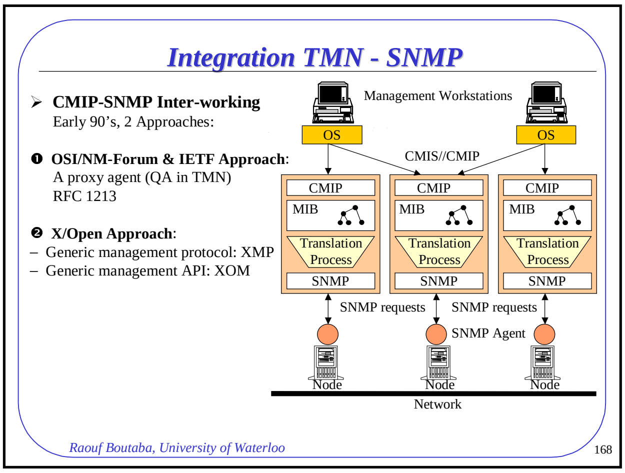


Network Management Forum Architecture

➢ **Key architectural concept : CME (Conformance Management Entity)**



CME Authority Relationships



Integration TMN - SNMP

➢ **CMIP-SNMP Inter-working**
Early 90's, 2 Approaches:

❶ **OSI/NM-Forum & IETF Approach**:
A proxy agent (QA in TMN)
RFC 1213

❷ **X/Open Approach**:
- Generic management protocol: XMP
- Generic management API: XOM

TMN integration within TINA

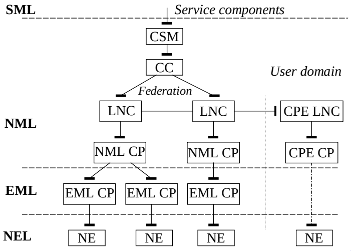
- **What is TINA?**
  - ☞ *In 1993, TINA-C (BellCore, NTT, BT, CSELT, Alcatel, Siemens, IBM, HP, …)*
  - ☞ **T**elecommunications **I**nformation **N**etworking **A**rchitecture
  - ☞ *Creation of telecom services, management of these services and the networks*
  - ☞ *Promote interoperability and reusability of telecommunication software*
  - ☞ **TINA** = IN + **TMN** + ODP
- **Overall view of the TINA architecture**
  - ☞ **Network**, **Service**, **Management** and **Computing** Architectures

TINA Architectures

- **TINA network architecture:**
  - ☞ **NRIM** (Network Resource Information Model)

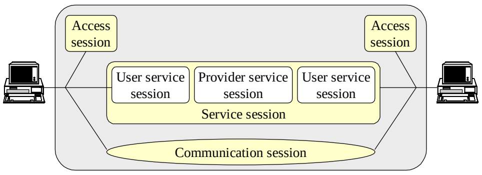
    Fragments: *Connection graph, Network, Connectivity, Termination point, Resource configuration, and Fault management fragments*
  - ☞ **Connection Management**

CSM: Comm. Session Mger
CC: Connection Coordinator
LNC: Layer Net. Coordinator
CP: Connection Performer
NE: Network Element
CPE: Customer Premises Net.
NEL: Network Element Layer
EML: Element Mgmt. Layer
NML: Net. Mgmt. Layer
SML: Service Mgmt Layer



TINA Architectures (Cont'd)

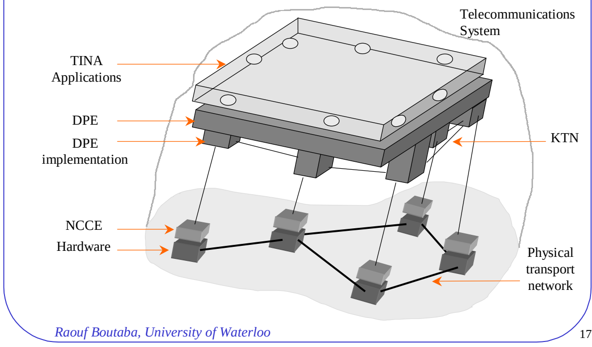
- **TINA service architecture:**



+ **USCM**: **U**niversal **S**ervice **C**omponent **M**odel

TINA Architectures (Cont'd)

- **TINA computing architecture: DPE** (*Distributed Processing Environment*)

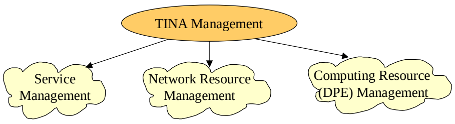


TINA Architectures (Cont'd)

- **TINA management architecture:**
  - ☞ Adopts the TMN for telecommunications network and service management:
    - TMN logical layered architecture, except

      Network Element + Network Element Management = Resource Management
    - TMN FCAPS, except

      Configuration management = resource configuration + Connection management
  - ☞ Adds distributed processing techniques
    - *Managing and managed systems as computational objects*


TINA Management
Service Management
Network Resource Management
Computing Resource (DPE) Management

Object Oriented Distributed Network Management - some Objectives

- Introduce mainstream distributed object technologies into the telecommunications management domain
- Use lower cost off the shelf products
- Integrate Telecommunications Management Information base with Enterprise Information base
- Reduce the specialized knowledge required to implement Telecommunications Management Systems
- Use Information technology solutions to software integration to resolve telecommunications software integration problems
- Take advantage of the advances in distributed systems technology in large scale integration/interworking of Telecoms Management Systems

Characteristics of “good” distributed systems

- **Resource sharing**
  - Hardware, data, applications
- **Openness**
  - Can the system be extended? Can new shared resources be added without disruption of existing resources? Open systems often provide uniform inter-process communication and published interfaces
  - Open systems can often be constructed with products from different vendors once conformance to some standard is adhered to and systems are properly certified and tested
- **Concurrency**
  - Many users efficiently interacting with a single threaded resource
  - One user efficiently interacting with multiple resources

Characteristics of “good” distributed systems

- **Scalability**
  - Increasing amount of data, increasing processing requirements, increasing number of users - need to maintain system/data integrity
- **Fault Tolerance**
  - Hardware redundancy
  - Software recovery
- **Transparencies**

– Access transparency	Failure transparency
– Location transparency	Migration transparency
– Concurrency transparency	Performance transparency
– Replication transparency	Scaling transparency

Challenges in building distributed systems

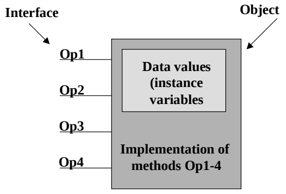
- **Naming**
  - Useful global meaning, efficient translation system, need to be able to scale
- **Communication**
  - Performance & reliability, interaction model, heterogeneous networks and type systems
- **Software structure**
  - Interface abstraction, scalability, granularity
- **Workload allocation**
  - Delegation of responsibilities to software elements to support changing performance requirements
- **Consistency maintenance**
  - Data integrity, cost of consistency

Characteristics of Object Oriented Programming

- **Object Identifier**
- **Object Operations**
- **Object Classes, Object Instances**
- **Inheritance**
- **Interface vs Implementation**

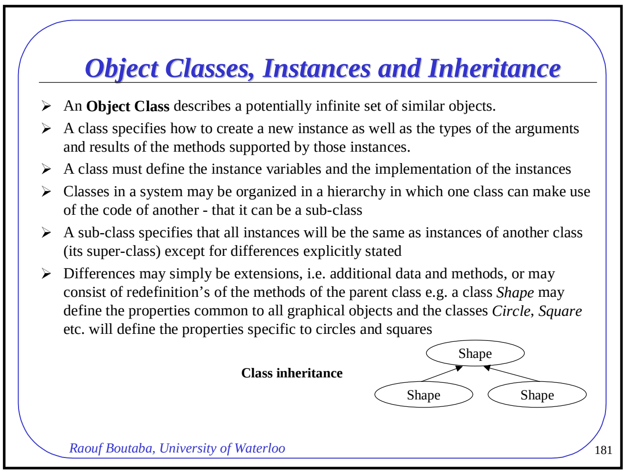
Object Oriented Programming languages

- *An object oriented program is usually described using an object oriented programming language e.g. C++, JAVA or Smalltalk*
- *An object provides services specified by its interface*
- *One can communicate with an object by sending it a message*
- *The message contains a request to perform one of the object operations*
- *An object contains data and specific instructions on how to perform its operations*
- *The specific instructions and data contained within an object are hidden from users of the object by the object interface*


Interface
Object
Op1
Op2
Op3
Op4
Data values (instance variables
Implementation of methods Op1-4

Object Identification/Operations

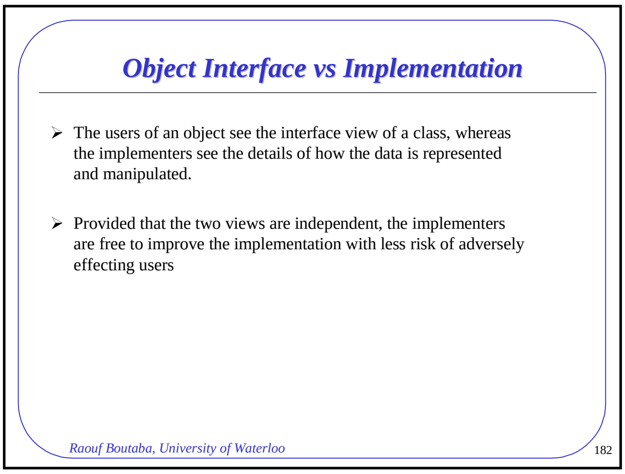
- **Object Identification**
- → Each object has an “Identifier”
- → Object identifiers can be passed by values, stored and/or returned as result of methods
- **Object Operations**
- → An object requiring some action to be performed sends a message to an object
- → That message results in the appropriate method invocation and (at some time defined by the object system) the return of control to the invoking object
- → A method invocation can result in one of or more of the following
  - ☞ further method invocations
  - ☞ a change in state of the object
  - ☞ further messages being sent to other objects
- → A Messages in object oriented systems request operation and can contain further information (parameters) needed to carry out the operation. Object interfaces define the format required of parameters and also the format of any values which may be returned to the requestor of a method invocation.



Object Classes, Instances and Inheritance

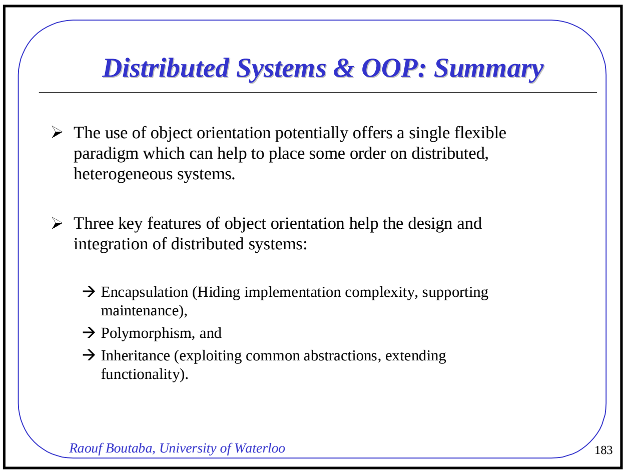
- An **Object Class** describes a potentially infinite set of similar objects.
- A class specifies how to create a new instance as well as the types of the arguments and results of the methods supported by those instances.
- A class must define the instance variables and the implementation of the instances
- Classes in a system may be organized in a hierarchy in which one class can make use of the code of another - that it can be a sub-class
- A sub-class specifies that all instances will be the same as instances of another class (its super-class) except for differences explicitly stated
- Differences may simply be extensions, i.e. additional data and methods, or may consist of redefinition's of the methods of the parent class e.g. a class *Shape* may define the properties common to all graphical objects and the classes *Circle*, *Square* etc. will define the properties specific to circles and squares

Class inheritance



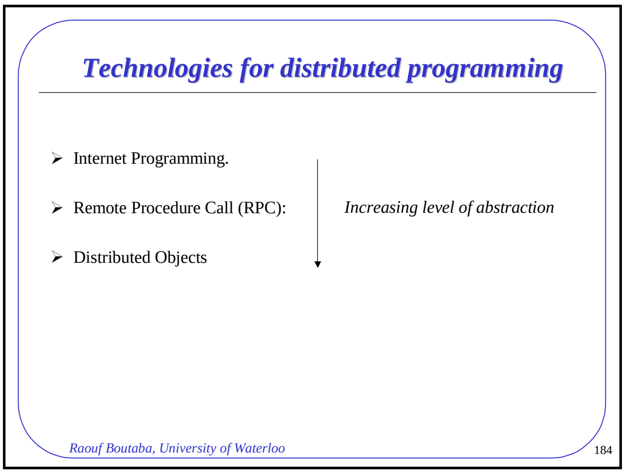
Object Interface vs Implementation

- The users of an object see the interface view of a class, whereas the implementers see the details of how the data is represented and manipulated.
- Provided that the two views are independent, the implementers are free to improve the implementation with less risk of adversely effecting users



Distributed Systems & OOP: Summary

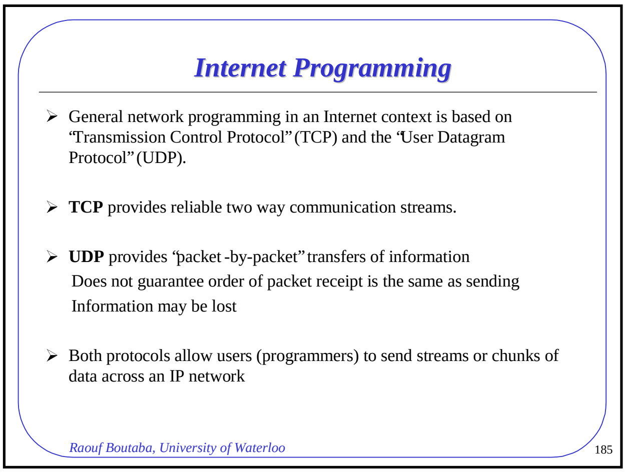
- The use of object orientation potentially offers a single flexible paradigm which can help to place some order on distributed, heterogeneous systems.
- Three key features of object orientation help the design and integration of distributed systems:
  - → Encapsulation (Hiding implementation complexity, supporting maintenance),
  - → Polymorphism, and
  - → Inheritance (exploiting common abstractions, extending functionality).



Technologies for distributed programming

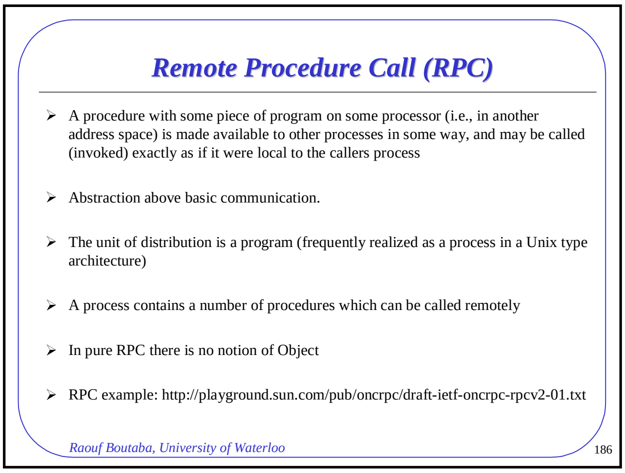
- Internet Programming.
- Remote Procedure Call (RPC):
- Distributed Objects

Increasing level of abstraction



Internet Programming

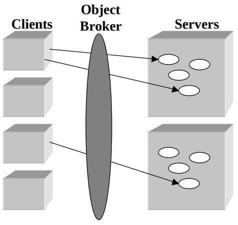
- General network programming in an Internet context is based on "Transmission Control Protocol"(TCP) and the "User Datagram Protocol"(UDP).
- **TCP** provides reliable two way communication streams.
- **UDP** provides "packet -by-packet"transfers of information
  Does not guarantee order of packet receipt is the same as sending
  Information may be lost
- Both protocols allow users (programmers) to send streams or chunks of data across an IP network



Remote Procedure Call (RPC)

- A procedure with some piece of program on some processor (i.e., in another address space) is made available to other processes in some way, and may be called (invoked) exactly as if it were local to the callers process
- Abstraction above basic communication.
- The unit of distribution is a program (frequently realized as a process in a Unix type architecture)
- A process contains a number of procedures which can be called remotely
- In pure RPC there is no notion of Object
- RPC example: http://playground.sun.com/pub/oncrpc/draft-ietf-oncrpc-rpcv2-01.txt

Distributed Objects

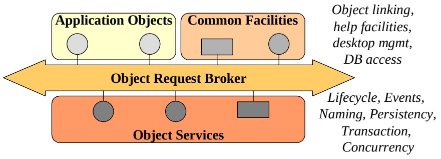


- ➢ Objects are logically grouped in servers
- ➢ Objects are accessible by clients
- ➢ Each object has well defined set of methods defined by its interface
- ➢ Servers are generally implemented as processes in modern operating systems
- ➢ An object broker is used to mediate between clients and objects
- ➢ An object can be invoked in the same way locally or remotely
- ➢ Applications can play the role of both Clients and Servers

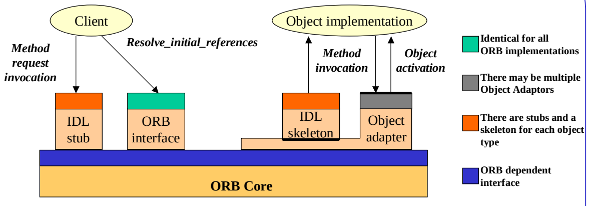
CORBA

Enable the development of distributed systems which support interoperability and portability based on an object oriented foundation which specifies:

- ➢ A single terminology for object orientation
- ➢ A common abstract framework or object model
- ➢ A common reference model or architecture
- ➢ Common interface and protocols


Application Objects
Common Facilities
Object Request Broker
Object Services
Object linking, help facilities, desktop mgmt, DB access
Lifecycle, Events, Naming, Persistency, Transaction, Concurrency

ORB Interfaces


Client
Method request invocation
Resolve_initial_references
IDL stub
ORB interface
Object implementation
Method invocation
Object activation
IDL skeleton
Object adapter
ORB Core
Identical for all ORB implementations
There may be multiple Object Adaptors
There are stubs and a skeleton for each object type
ORB dependent interface

- ➢ **Dynamic Invocation**
  a client may dynamically construct and invoke requests on objects
- ➢ **Client Stub**
  represents a possible object operation (language dependent)
- ➢ **ORB interface**
  interface to ORB operations common to all objects, e.g. return object's interface type
- ➢ **Implementation skeleton**
  interface through which an object-method is invoked
- ➢ **Object Adaptor**
  access to services such as activation, deactivation, object reference management, object creation, ...

CORBA Services

- ➢ Naming Service
- ➢ Event Management Service
- ➢ Persistent Object Service
- ➢ Lifecycle Service
- ➢ Concurrency Service
- ➢ Transaction Service
- ➢ Query Service
- ➢ Security Service
- ➢ Time Service
- ➢ Relationships Service
- ➢ Licensing Service
- ➢ Trader Service
- ➢ Collection Service
- ➢ ...

OMG Interface Definition Language (IDL)

Supports the definition of Objects which in turn support methods which can be provided and accessed via a CORBA implementation

OMG IDL Separates the Interface from the Implementation:

- ➢ multiple-inheritance, strongly typed, public interface specification language
- ➢ independent of any particular language/compiler
- ➢ mappings will be provided for many languages/compilers
- ➢ not a programming language

Enables Interoperability

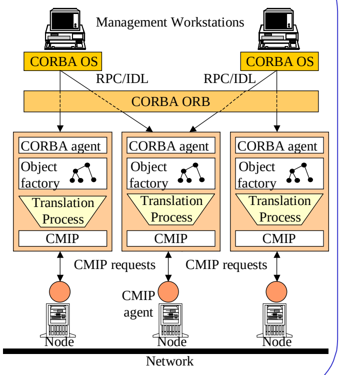
Simple Example OMG IDL Interface

```
Module SimpleStocks {
          interface StockMarket
          {
                    float       get_price {in string symbol};
          };
};
```

If I create a CORBA Object Instance on my computer and send someone an appropriate 'CORBA Object Reference" to this instance. If they h ave the definition above they should be able to call the get_price method passing the method a string and get a result returned.

CORBA support for the TMN

- **CORBA to provide DPE services for TMN**
  - ☞ Messaging service
  - ☞ Naming service
  - ☞ Notification service
  - ☞ Info-Model service
- **OSI/CORBA interoperability in the TMN framework**
  - ☞ IDL from/to GDMO/ASN.1
  - ☞ CORBA msg. f/t CMIP PDUs
- **CORBA-OSI/CMISE Gateway**
  - ☞ X/Open-JIDM task force (*Joint Inter-Domain Management*)
  - ☞ OSI/NM-Forum
  - ☞ OMG-TSIG (*Telecom Special Interest Group*)



TeleMangement Forum

- **TM Forum**

  Provides Leadership, strategic guidance and practical solutions to improve the management and operation of communications services
- **TM Forum Approach**

  Business and customer services driven approach

  Based on the business layering principles articulated in the ITU-T layered TMN model
- **TM Forum Programs**
  - ***Process Automation Program***
    - Telecom Operation Map (**TOM**)
  - ***Technology Integration Program***
    - Technology Integration Map (**TIM**)

      appropriate technologies and how should be the integration
    - Central Information Facility (**CIF**)

      web-based TM Forum Repository
  - ***Catalyst Projects***
    - Implementations process automation solutions

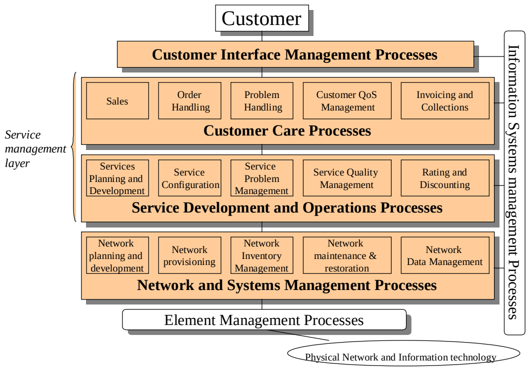
TOM: Telecom Operation Map

- ***Motivation***
  - *Service providers face very different regulatory environments and their business strategies and approaches to competition are quite distinct*
  - *They share several common characteristics*
- ***Objectives***
  - *Establishing common specifications*
- ***Approach***
  - Identifying the *business objectives* and *business process framework*
    - ***An 'industry owned" common business process model***
    - ***Common definitions to describe processes of a service provider***
    - ***Agreement on the basic information***

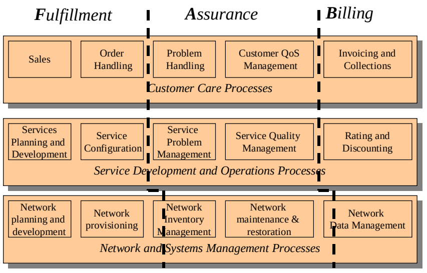
      required to perform each process, sub-process and process activity
    - ***A process framework***

      for identifying which *process* and *interfaces* are in most need of integration and automation, and most dependent on industry agreement

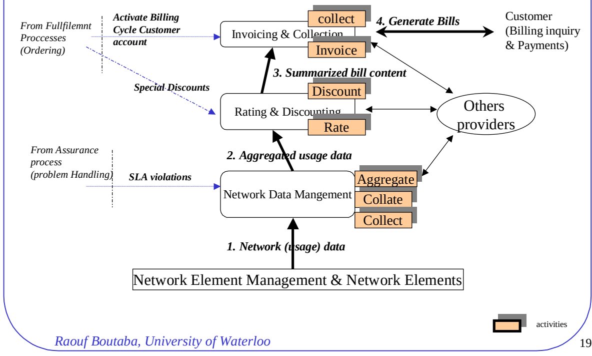
Tom, Business Process Framework


Customer
Customer Interface Management Processes
Sales
Order Handling
Problem Handling
Customer QoS Management
Invoicing and Collections
Customer Care Processes
Service management layer
Services Planning and Development
Service Configuration
Service Problem Management
Service Quality Management
Rating and Discounting
Service Development and Operations Processes
Network planning and development
Network provisioning
Network Inventory Management
Network maintenance & restoration
Network Data Management
Network and Systems Management Processes
Element Management Processes
Physical Network and Information technology
Information Systems management Processes

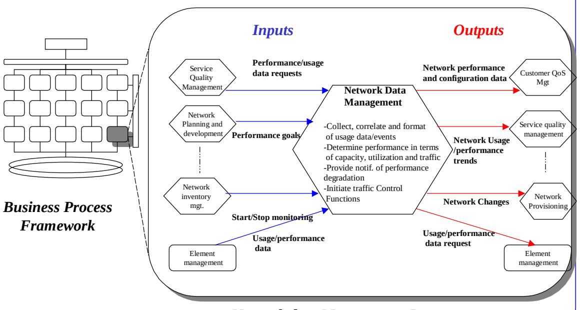
TOM, FAB Business Process Breakdown


Fulfillment
Assurance
Billing
Sales
Order Handling
Problem Handling
Customer QoS Management
Invoicing and Collections
Customer Care Processes
Services Planning and Development
Service Configuration
Service Problem Management
Service Quality Management
Rating and Discounting
Service Development and Operations Processes
Network planning and development
Network provisioning
Network inventory Management
Network maintenance & restoration
Network Data Management
Network and Systems Management Processes

Examples of Billing Process Flow



TOM: Example of an Operational Process


Business Process Framework
Inputs
Outputs
Service Quality Management
Network Planning and development
Network inventory mgt.
Element management
Performance/usage data requests
Performance goals
Start/Stop monitoring
Usage/performance data
Network Data Management
-Collect, correlate and format of usage data/events
-Determine performance in terms of capacity, utilization and traffic
-Provide notif. of performance degradation
-Initiate traffic Control Functions
Network performance and configuration data
Network Usage /performance trends
Network Changes
Usage/performance data request
Customer QoS Mgt
Service quality management
Network Provisioning
Element management

Network data Management Process

Catalyst Projects

- Products solutions

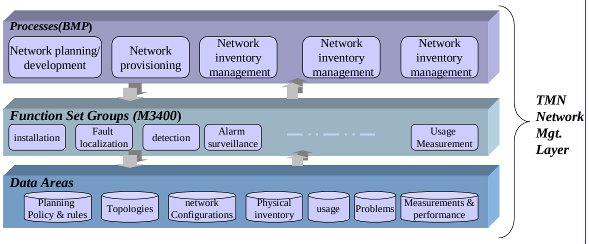
  Catalyst Projects are intended to kick-start the industry in specific areas by linking together existing products to meet a specific market need

- TMF Catalyst projects
  - ✥ Service Fulfillment Program
    - ATM Management
    - Connection and Service Management Catalyst
    - IP Service Management
    - SONET/SDH/DWDM Management
  - ✥ Service Fulfillment Program
    - Internet Customer Care
    - Mobile Service Quality Management
    - Plug & Play End-to-End Service Assurance Catalyst
    - SLA Management Catalyst

TM-Forum & Network Management

✥ Network Management Detailed Operations Map

Provides Network Management Processes and Functions

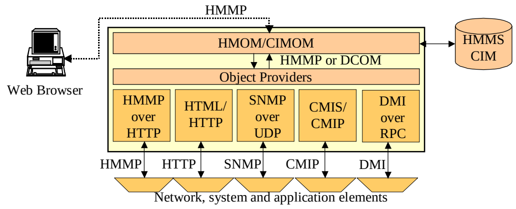


Internet Technologies for Network Management

- **Web-based Network Management**
- **Network Management in Java**
- **Software Mobile Agents in Network Management**
- **Active Networks for Programmable Management**

Web-Based Enterprise Management

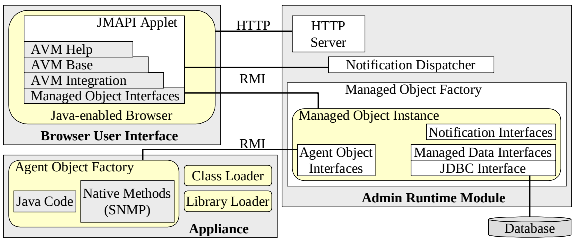
- **What is WBEM (Web-Based Enterprise Management) ?**
  - ✥ *to manage disparate networks, systems & applications through Web-browsers*
  - ✥ Cooperative effort of Microsoft, Intel, BMC Software, Compaq & 75 others
  - ✥ April 1997 adopted as a standard by DMTF (Desktop Management Task Force)
- **WBEM Basics:**



Java-based Management

- **An Example is JMAPI**
  - ✥ *a product of SUN but also involved CISCO, Novell, Bay Networks and others*
- **What is JMAPI (Java Management API)?**
  - ✥ *To provide reusable management-specific Java classes*
  - ✥ *To develop Web-based object-oriented management applications*
  - ✥ *To implement distributed management using RMI (Remote Method Invocation)*
  - ✥ *To allow for platform-independent management using JVM (Java Virtual Machine)*

JMAPI Architecture

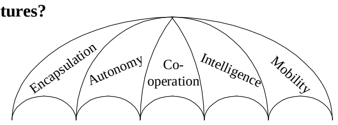


Software Agents

- **What is an Agent?**

  A self-contained software element responsible for performing part of a programmatic process

- **Agents' features?**

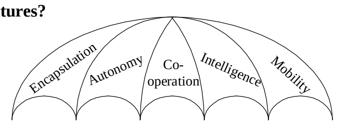


Software Agents for Network Management

- **What is an Agent?**

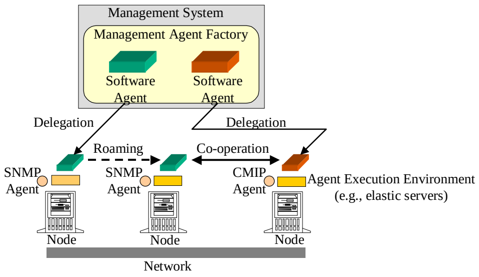
  A self-contained software element responsible for performing part of a programmatic process

- **Agents' features?**



- **Why Agents in network management?**
  - *to solve problems such as scalability, latency, delays*
  - *to automate control and management processes*
  - *to allow for network programmability*
  - *to allow for rapid provision of new and customized network services*

TMN & Agent Technologies



Policy-based Networking/Management

- **Policy-based Networking/Management : A hot topic !**

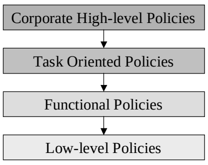
  Policy servers implemented by CISCO, 3COM, Bay Networks, Cabletron, ...

- **What is a Policy?**
  - *the plan of an organization to achieve its goals*
  - *General rule set governing network operation and service deployment*

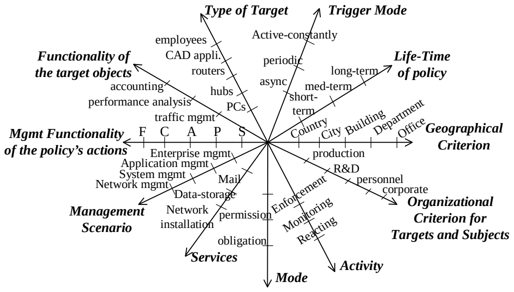
- **Policy Representation?**

```
Policy_id
mode
[condition]
subject
{action}
target
[when constraint];
```

Policy Hierarchies:



Policy Classification



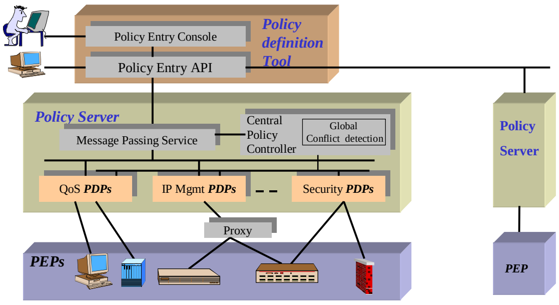
Motivations for Policy-based Net/Man

- *Enable intelligent, environment-based access to and control of network resources*
- *Improve network management (especially device configuration and provisioning*
- *Provide personalized network services*

Policy-based Net/Man Architecture

- **Functional Requirements**
  - *Enforcement (mechanism)*
  - *Decision making*
  - *Policing (on-going action)*
- **Architecture Components**
  - ***Policy Definition Tool***
    A centralized tool, where policies are defined, edited, and managed.
  - ***Policy Server***
    Policy Decision Point ***PDP*** *that controls the application of configuration changes*
  - ***Policy Enforcement Point PEP***
    Enforces policies
    Communicates with PDP (different protocols are possible, but COPS defined)
    A proxy may be used between PDP and PEP if PEP is not policy-capable

Example Architecture



COPS-Common Open Policy Service

- **What is COPS ?**
  - **COPS Service:**
    A client/ server model for supporting policy control
  - **COPS Protocol:**
    A query response protocol used to exchange policy information between a network policy server and a set of clients
  - **Being developed within IETF/RAP-WG (RSVP Admission Policy WG)**
    Originally, COPS was associated with Resource Reservation Protocol (RSVP) as mechanism to allow devices to look up external information. (QoS Policy)
  - **Being extended to be…**
    Used for Differentiated Services IP
    Support diverse client specific information
    Support other network services such as security and multicast

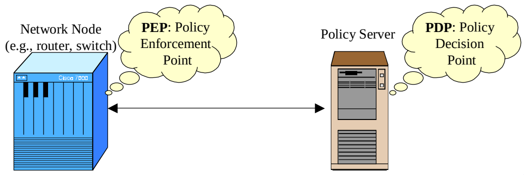
COPS-Common Open Policy Service

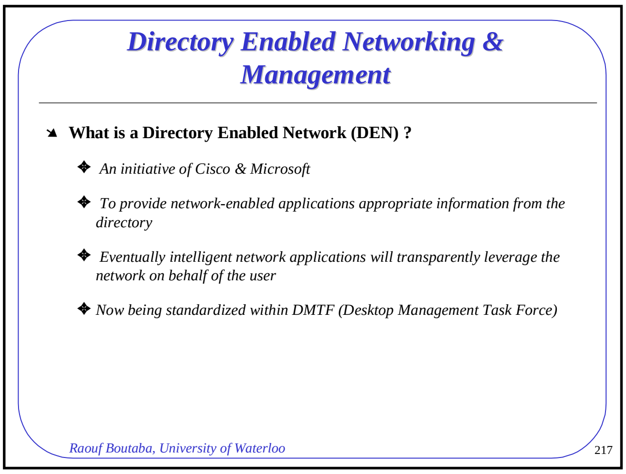
- **Policy Modeling in COPS**
  - *Objects-oriented*
  - *Uniquely identified with* ***PIDs*** *(Policy IDentifier)*
  - *Tree-like structured :*

    *policy classes (****PCs****) as the nodes and Policy Instances (****PIs****) as the leaves*
  - *Stored in a database: Policy Information Base (****PIB****)*

COPS-Common Open Policy Service

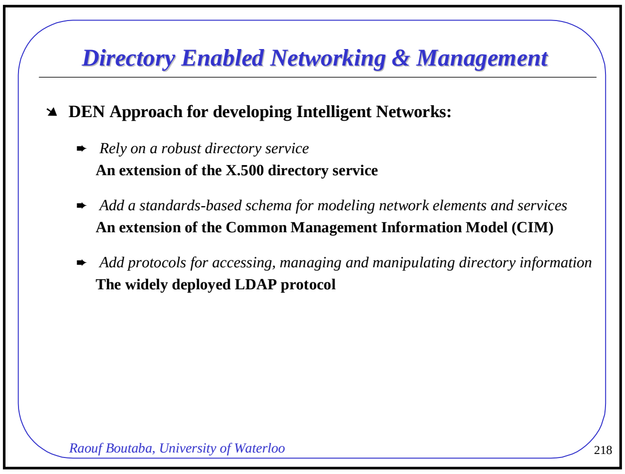
- **Policy Control in COPS**
  - *Initial Request from the PEP to begin a manager agent session for policy*
  - *'Client type" filed, in the COPS message, allows different PEP/ PDP pairs to communicate different kinds of policy using the same basic system.*
  - *PEP queries the PDP about specific client objects, PDP returns the appropriate information*
  - *PDP revokes or updates assigned policy if conditions change.*





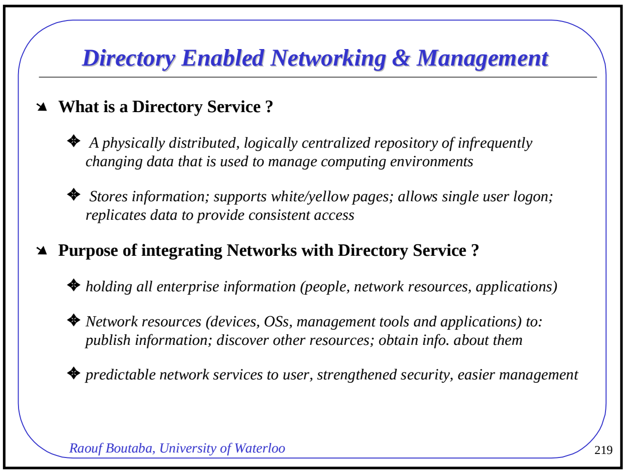
Directory Enabled Networking & Management

- What is a Directory Enabled Network (DEN) ?
  - *An initiative of Cisco & Microsoft*
  - *To provide network-enabled applications appropriate information from the directory*
  - *Eventually intelligent network applications will transparently leverage the network on behalf of the user*
  - *Now being standardized within DMTF (Desktop Management Task Force)*



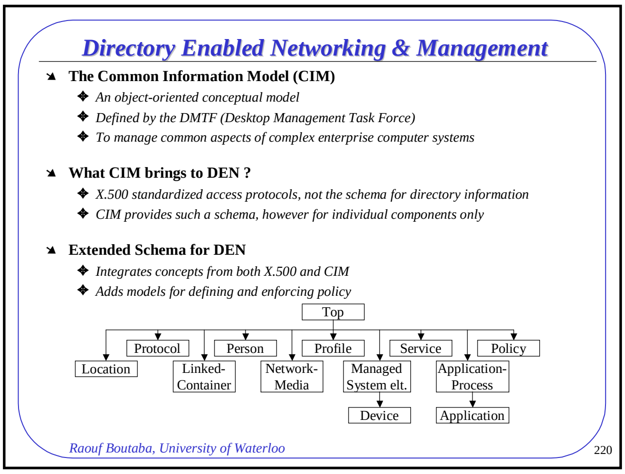
Directory Enabled Networking & Management

- **DEN Approach for developing Intelligent Networks:**
  - *Rely on a robust directory service*
    An extension of the X.500 directory service
  - *Add a standards-based schema for modeling network elements and services*
    An extension of the Common Management Information Model (CIM)
  - *Add protocols for accessing, managing and manipulating directory information*
    The widely deployed LDAP protocol



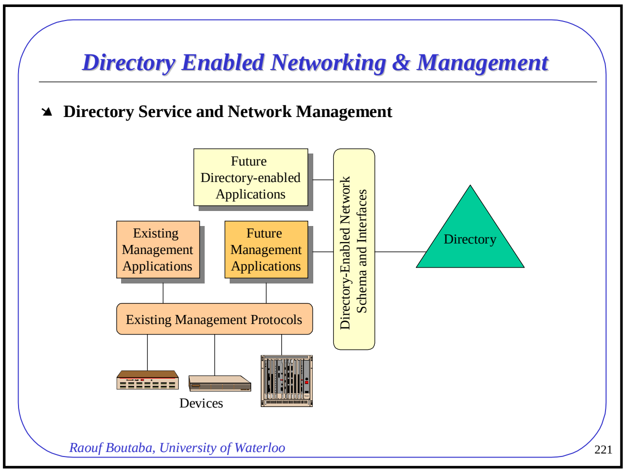
Directory Enabled Networking & Management

- **What is a Directory Service ?**
  - *A physically distributed, logically centralized repository of infrequently changing data that is used to manage computing environments*
  - *Stores information; supports white/yellow pages; allows single user logon; replicates data to provide consistent access*
- **Purpose of integrating Networks with Directory Service ?**
  - *holding all enterprise information (people, network resources, applications)*
  - *Network resources (devices, OSs, management tools and applications) to: publish information; discover other resources; obtain info. about them*
  - *predictable network services to user, strengthened security, easier management*



Directory Enabled Networking & Management

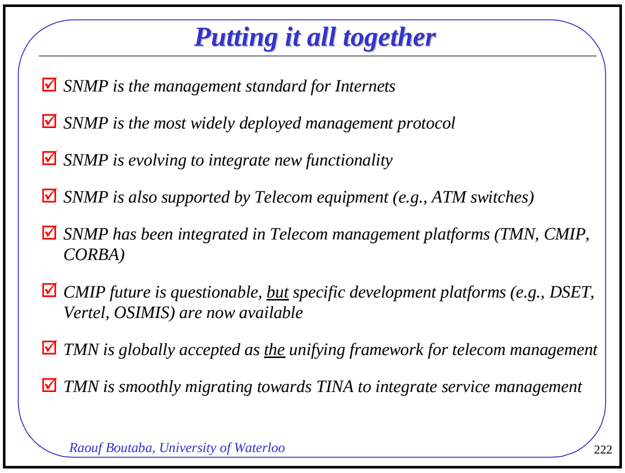
- **The Common Information Model (CIM)**
  - *An object-oriented conceptual model*
  - *Defined by the DMTF (Desktop Management Task Force)*
  - *To manage common aspects of complex enterprise computer systems*
- **What CIM brings to DEN ?**
  - *X.500 standardized access protocols, not the schema for directory information*
  - *CIM provides such a schema, however for individual components only*
- **Extended Schema for DEN**
  - *Integrates concepts from both X.500 and CIM*
  - *Adds models for defining and enforcing policy*



Directory Enabled Networking & Management

- **Directory Service and Network Management**

Future Directory-enabled Applications
Existing Management Applications
Future Management Applications
Existing Management Protocols
Devices
Directory-Enabled Network Schema and Interfaces
Directory



Putting it all together

☑ *SNMP is the management standard for Internets*

☑ *SNMP is the most widely deployed management protocol*

☑ *SNMP is evolving to integrate new functionality*

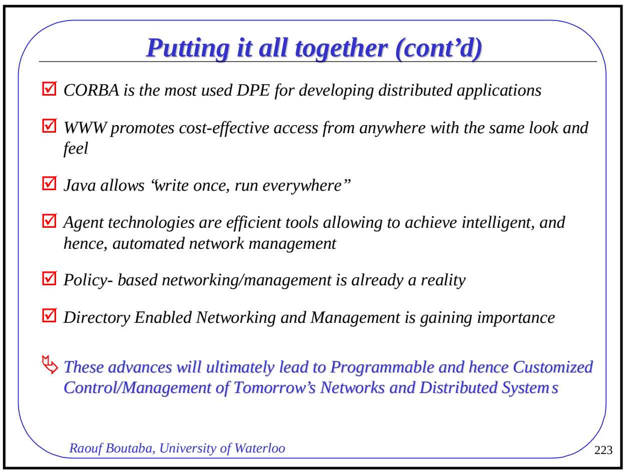
☑ *SNMP is also supported by Telecom equipment (e.g., ATM switches)*

☑ *SNMP has been integrated in Telecom management platforms (TMN, CMIP, CORBA)*

☑ *CMIP future is questionable, but specific development platforms (e.g., DSET, Vertel, OSIMIS) are now available*

☑ *TMN is globally accepted as the unifying framework for telecom management*

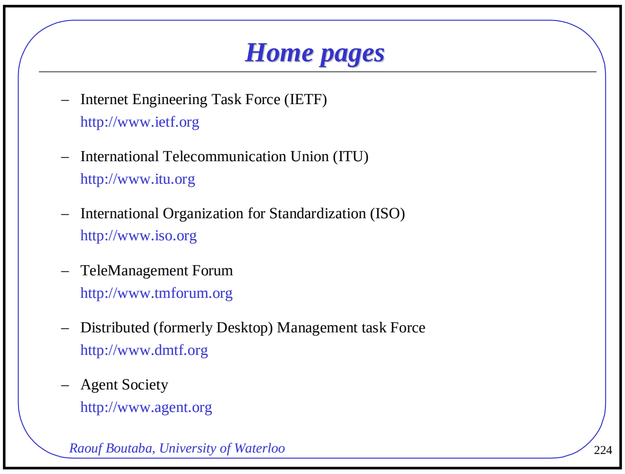
☑ *TMN is smoothly migrating towards TINA to integrate service management*



Putting it all together (cont'd)

- ☑ *CORBA is the most used DPE for developing distributed applications*
- ☑ *WWW promotes cost-effective access from anywhere with the same look and feel*
- ☑ *Java allows "write once, run everywhere"*
- ☑ *Agent technologies are efficient tools allowing to achieve intelligent, and hence, automated network management*
- ☑ *Policy- based networking/management is already a reality*
- ☑ *Directory Enabled Networking and Management is gaining importance*

➯ *These advances will ultimately lead to Programmable and hence Customized Control/Management of Tomorrow's Networks and Distributed Systems*



Home pages

– Internet Engineering Task Force (IETF)
http://www.ietf.org

– International Telecommunication Union (ITU)
http://www.itu.org

– International Organization for Standardization (ISO)
http://www.iso.org

– TeleManagement Forum
http://www.tmforum.org

– Distributed (formerly Desktop) Management task Force
http://www.dmtf.org

– Agent Society
http://www.agent.org