

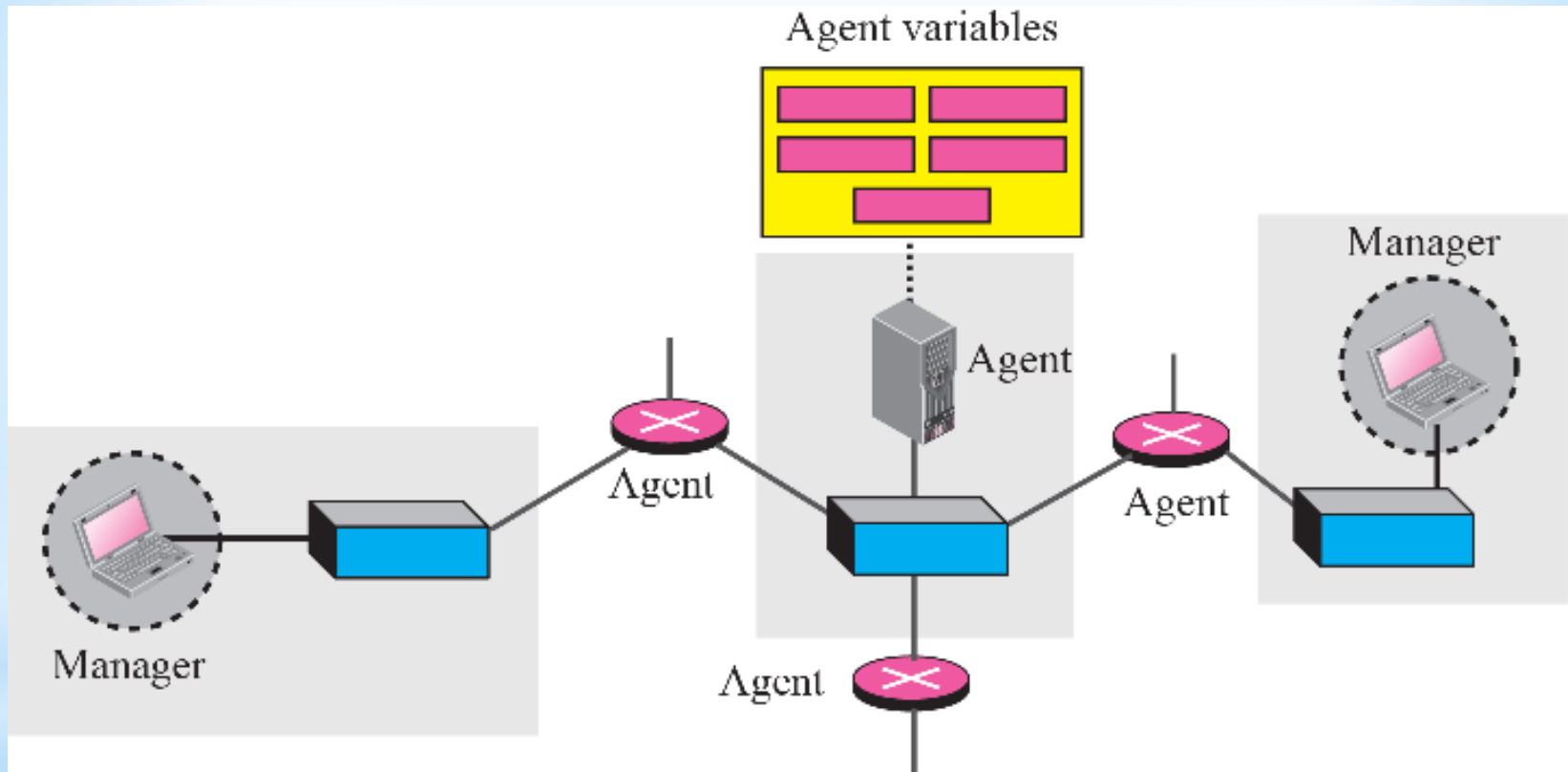
Network Management

Dr. Yasir Ahmed

**Baghdad University
Al-Khwarizmi Engineering College
Information and Communication**

2018

Recap of Previous Lecture

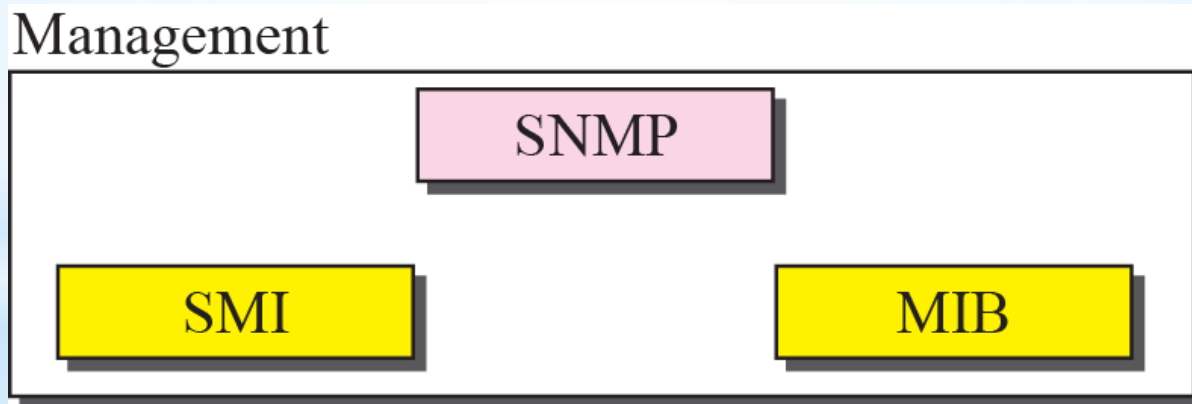


Recap of Previous Lecture

- SNMP uses the concept of manager and agent. That is, a manager, usually a host, controls and monitors a set of agents, usually routers or servers.
- SNMP is an application program that allows:
 1. A manager to retrieve the value of an object defined in an agent.
 2. A manager to store a value in an object defined in an agent.
 3. An agent to send an alarm message about an abnormal situation to the manager.

SNMP Framework Components

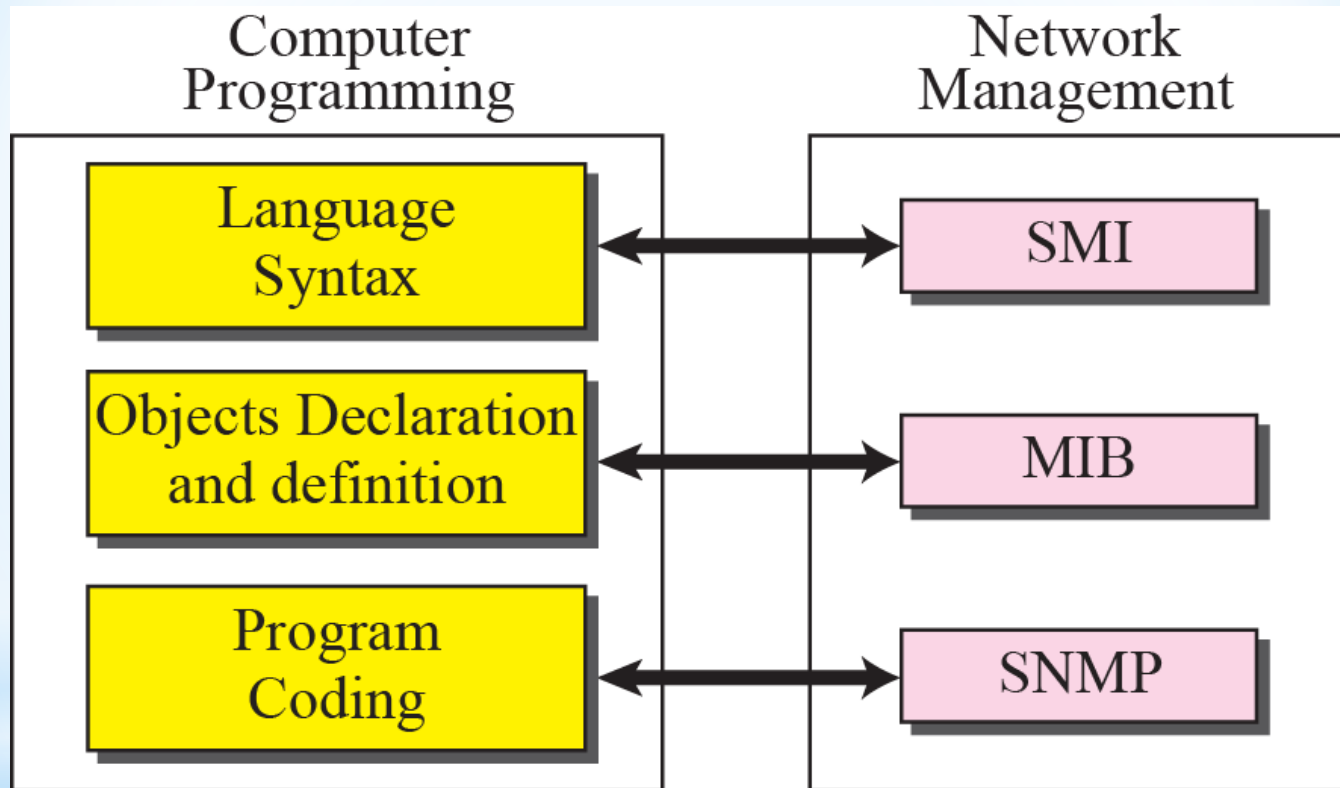
To do management tasks, SNMP uses two other protocols: **Structure of Management Information (SMI)** and **Management Information Base (MIB)**. In other words, management on the Internet is done through the cooperation of three protocols: SNMP, SMI, and MIB



SNMP Framework Components

- SNMP defines the format of packets exchanged between a manager and an agent. It reads and changes the status of objects (values of variables) in SNMP packets.
- SMI defines the general rules for naming objects, defining object types (including range and length), and showing how to encode objects and values.
- MIB creates a collection of named objects, their types, and their relationships to each other in an entity to be managed.

Cooperation between Protocols



Information Model

Structure of Management Information (SMI)

- SMI defines for a managed object
 - Syntax
 - Semantics – i.e., definition
 - plus additional information such as status
- Example
 - sysDescr: { system 1 }
 - Syntax: OCTET STRING
 - Definition: "A textual description of the entity. "
 - Access: read-only
 - Status: mandatory

Information Model

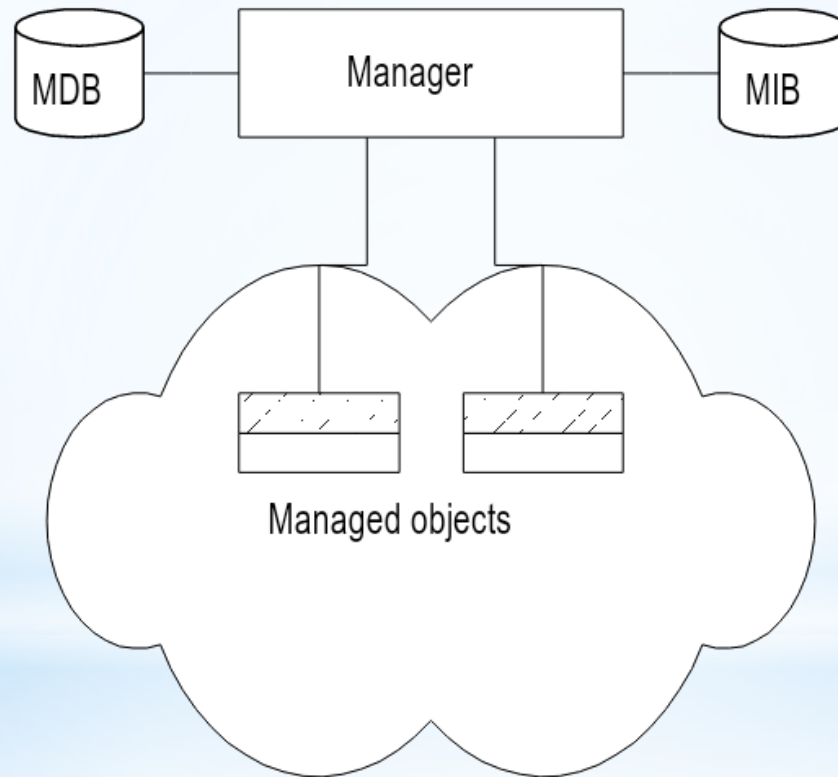
Management Information Base (MIB)

- Used by manager & agents to store & exchange management information
- Information base contains information about objects organized by grouping of related objects (e.g., IP group)
- Defines relationship between objects (e.g., object system is a parent of object sysDescr)
- It is NOT a physical database. It is a virtual database that is compiled into management module
- The agent MIB is used for accessing local information requested by the manager, and sending a response back
- The manager MIB is used for accessing information on all network components the manager manages.

Information Model

- **Management DataBase / Information Base**

Distinction between MDB and MIB



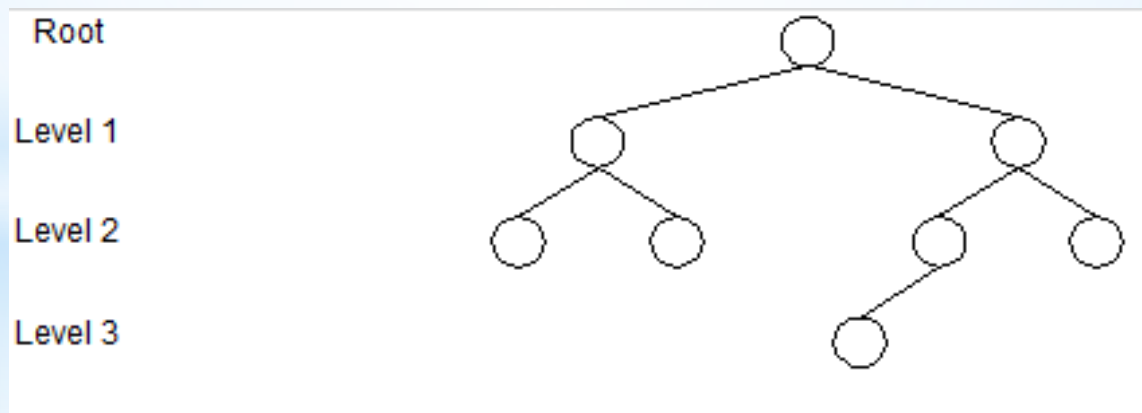
Information Model

Distinction between MDB and MIB

- MDB is a real database and contains the measured or administratively configured values of the elements of the network.
- MIB is a virtual database and contains the information necessary for processes to exchange information.
- An NMS can automatically discover a managed object, such as a hub, when added to the network.
- The NMS can identify the new object as hub only after the MIB information on the hub is added to the manager's MIB. This information is actually compiled into the manager's MIB schema.

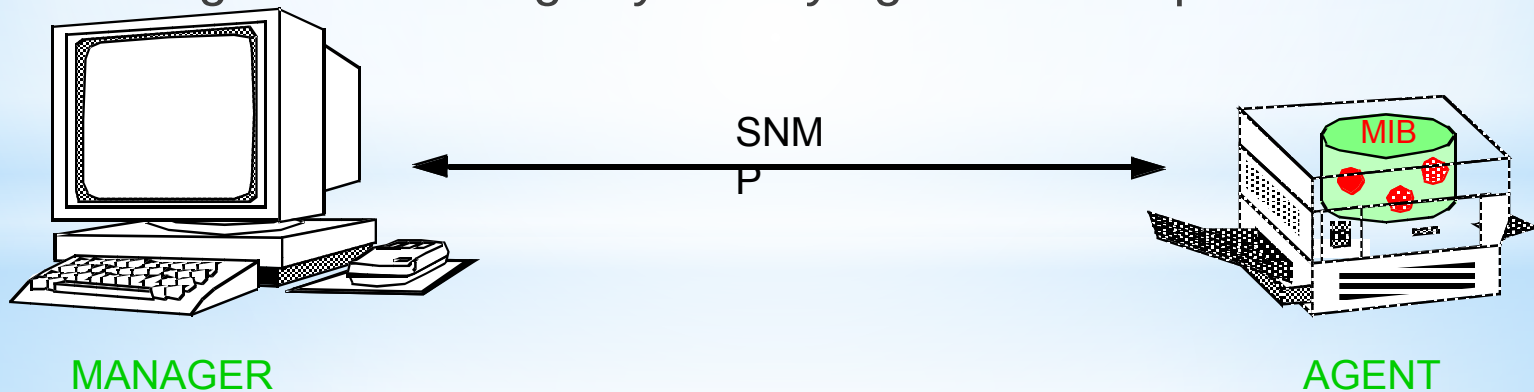
Management Information Trees

- Managed objects are uniquely defined by a tree structure specified by the OSI model and are used in the Internet model.
- Each managed object occupies a node in the tree.
- The hierarchical naming scheme is useful for providing a naming structure and avoiding name collisions.



Communication Model

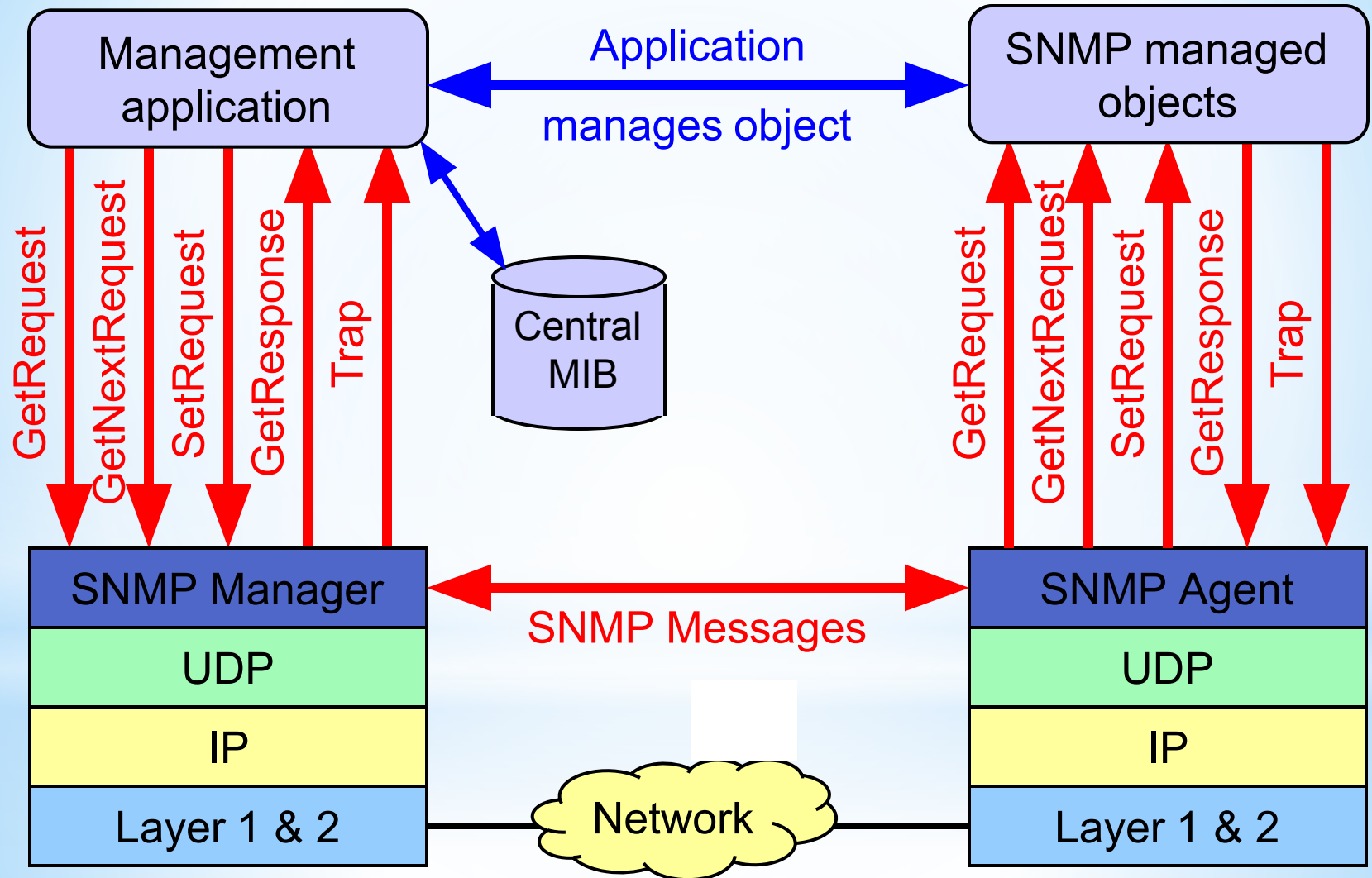
- Resources are represented as objects (or data variables)
 - Collection of objects is a MIB (more later)
 - A manager performs monitoring by retrieving the value of MIB objects
 - A manager causes an action to take place or changes the configuration settings by modifying values of specific variables



Communication Model

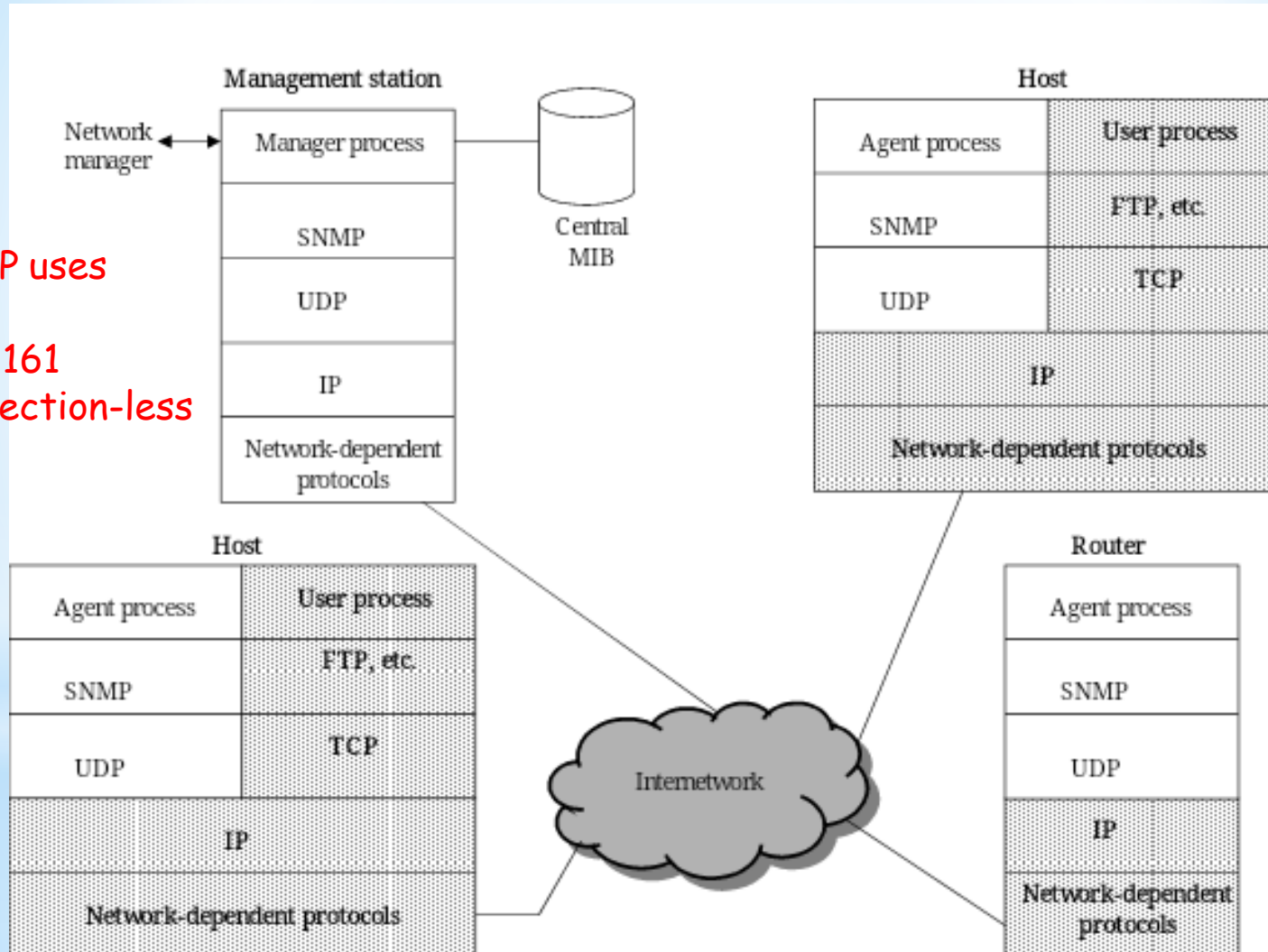
- The communication model is associated with the way information is exchanged between systems.
- Three aspects need to be addressed in the communication of information between two entities:
 - Transport medium of message exchange (transport protocol),
 - Message format of communication (application protocol)
 - Actual message(commands and responses).

Communication Model



Communication Model - Protocol Architecture

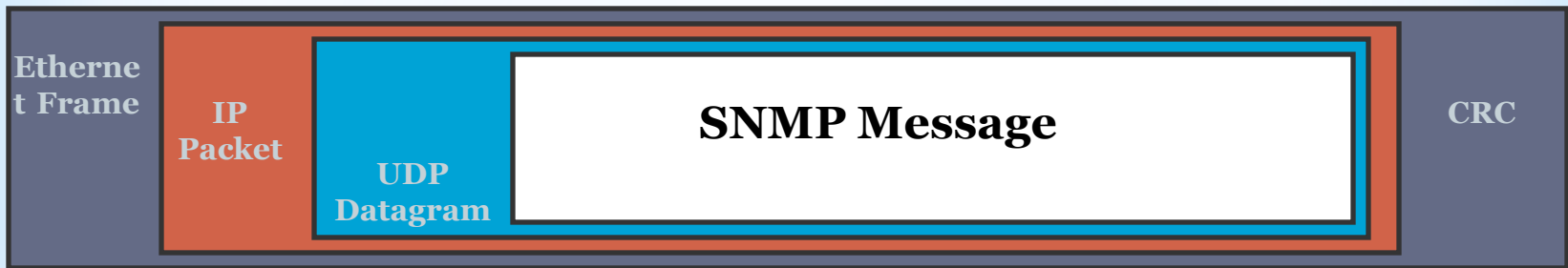
- SNMP uses UDP
- port 161
- connection-less



Interprets SNMP messages and controls the agent's MIB

Ports & UDP

- SNMP uses User Datagram Protocol (UDP) as the transport mechanism for SNMP messages



- Like FTP, SNMP uses two well-known ports to operate:
- **UDP Port 161** - SNMP Messages
- **UDP Port 162** - SNMP Trap Messages

Manager and Agent Interaction

Interactions between the SNMP manager and managed network elements (SNMP agents) can be any of three different types of commands:

- Read— To monitor managed network elements, SNMP managers read MIB variables maintained by the SNMP agents.
- Write— To control managed network elements, SNMP managers write MIB variables stored within the SNMP agents.
- Notification— SNMP agents use traps or informs to asynchronously report certain events to SNMP managers. Traps are unacknowledged notifications, and informs are notifications acknowledged by the SNMP manager.

Client Pull & Server Push

- **SNMP is a “client pull” model**

The management system (client) “pulls” data from the agent (server).

- **SNMP is a “server push” model**

The agent (server) “pushes” out a trap message to a (client) management system

SNMP Versions

- SNMP version 1 (SNMPv1)
- SNMP version 2 (SNMPv2)
- SNMP version 3 (SNMPv3)

SNMP version 1 (SNMPv1)

SNMPv1 is a simple request/response protocol that specifies five SNMP operations, also **called Protocol Data Unit (PDU)**:

- **Get**— Allows the SNMP manager to retrieve an object instance from the agent.
- **Get-Next**— Allows the SNMP manager to retrieve the next object instance from a table or list within an SNMP agent.
- **Set**— Allows the SNMP manager to set values for object instances within an SNMP agent.
- **Trap**— Used by the SNMP agent to asynchronously notify the SNMP manager of some event.
- **Response**— Used as a response for the Get and Set.

SNMPv1 does not formally define a functional model

- What was the functional model?
- Deals with the user oriented requirements: (configuration, fault, performance, security, and accounting)

SNMPv1 - Traps

- Traps are unrequested event reports that are sent to a management system by an SNMP agent process
- When a trappable event occurs, a trap message is generated by the agent and is sent to a trap destination (a specific, configured network address)
- Many events can be configured to signal a trap, like a network cable fault, failing NIC or Hard Drive, a “General Protection Fault”, or a power supply failure
- Traps can also be throttled -- You can limit the number of traps sent per second from the agent
- Traps have a priority associated with them -- Critical, Major, Minor, Warning, Marginal, Informational, Normal, Unknown

SNMPv1 - Trap Receivers

- Traps are received by a management application.
- Management applications can handle the trap in a few ways:
 - Poll the agent that sent the trap for more information about the event, and the status of the rest of the machine.
 - Log the reception of the trap.
 - Completely ignore the trap.

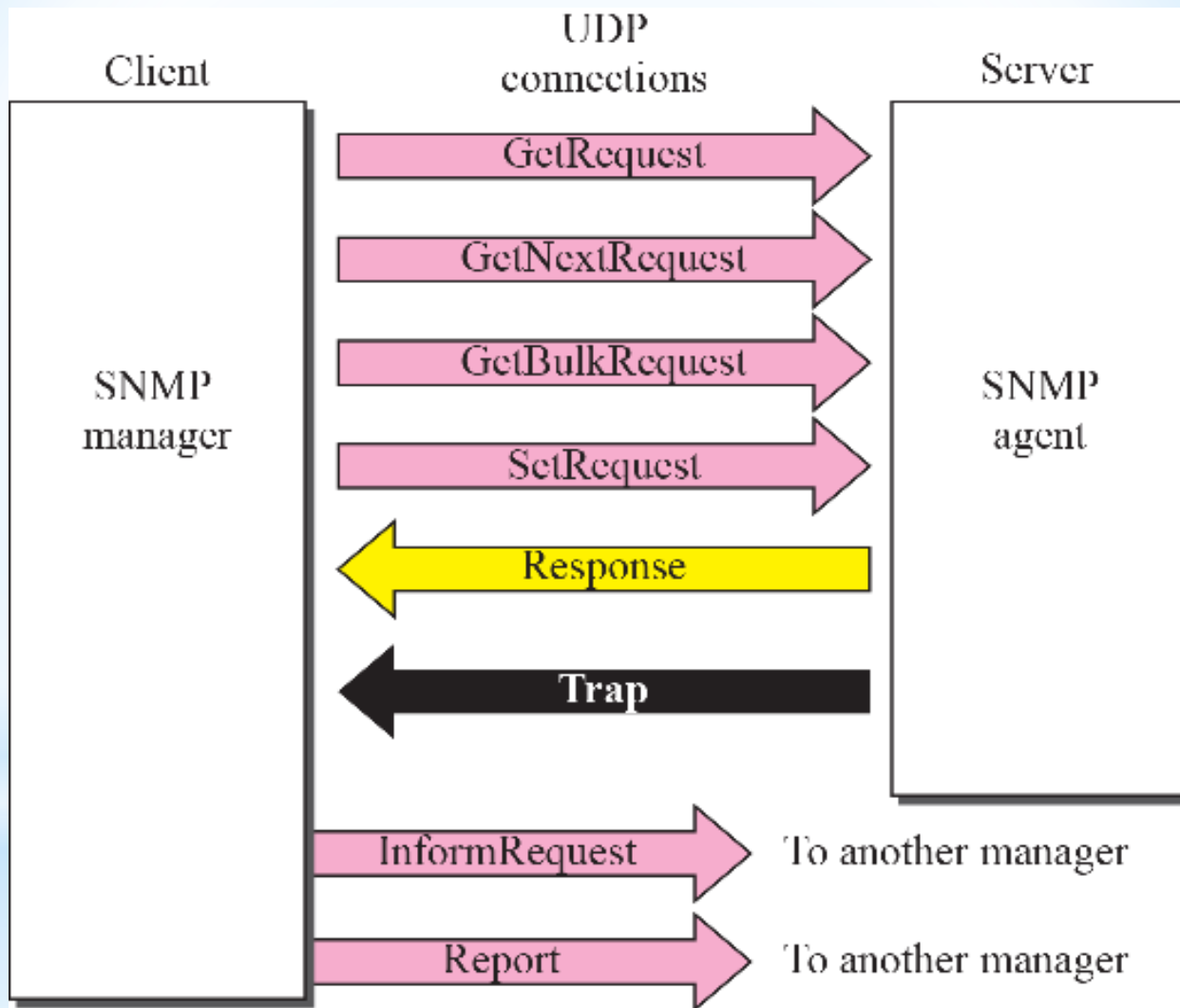
SNMP version 2 (SNMPv2)

Even though SNMPv2 has no improved security compared to SNMPv1, it offers some advantages, thanks to the SNMPv2 Framework:

- Richer error handling (error-status: wrongType, wrongLength, wrongEncoding, etc.)
- New data types (for example, 64-bit counters)
- New PDUs:

- New PDUs:
 - GetBulk - The GetBulk operation was added to make it easier to acquire large amounts of related information without initiating repeated get-next operations. As an example, retrieving a routing table of 100 entries from a network element requires one GetBulk operation in SNMPv2 compared to 101 GetNext PDUs in SNMPv1 (the 101st operation indicates the end of the row).
 - Inform - The Inform operation was added to allow the acknowledgment of notification, because one of the drawbacks of the trap is its unreliability due to the UDP transport protocol. The remote manager responds with a Response PDU.
 - Trap - The trap operation has been redesigned in SNMPv2.

SNMPv2



SNMP version 3 (SNMPv3)

- The SNMP version 3 (SNMPv3) Management Framework addresses the deficiencies in SNMPv2 related to security and administration.
- SNMPv3 provides secure access to network elements through a combination of authenticating and encrypting packets over the network.

Here are the security features provided in SNMPv3:

- Message integrity— Ensuring that a packet has not been tampered with in transit.
- Authentication— Determining that the message is from a valid source.
- Encryption— Encrypting the content of a packet prevents eavesdropping on the exchanges between the SNMP manager and agent.
- Message stream modification— Avoids malicious reordering, delay, or replay of messages.
- To make the security system easy to configure or change, v3 allows *remote configuration*, meaning that an authorized manager can change the configuration of security items listed above without being physically present at the device.