

# Computer Networks(2015 Pattern)

## Unit VI – Application Layer

**By Prof. A.R.Jain**

**PVG's COE,Nashik**

**Note: Material for this presentations are taken from Internet and books and only being used for student reference**

9/19/2017

# Outline

**Hyper Text Transfer Protocol (HTTP),**

FTP,

Email: SMTP, MIME, POP3, Webmail,

Domain Name System (DNS),

TELNET,

Dynamic Host Control Protocol (DHCP),

Simple Network Management Protocol (SNMP).

# Web and HTTP

## First some jargon

- **Web page** consists of **objects**
- Object can be HTML file, JPEG image, Java applet, audio file,...
- Web page consists of **base HTML-file** which includes several referenced objects
- Each object is addressable by a **URL**
- Example URL:

`www.someschool.edu/someDept/pic.gif`

host name

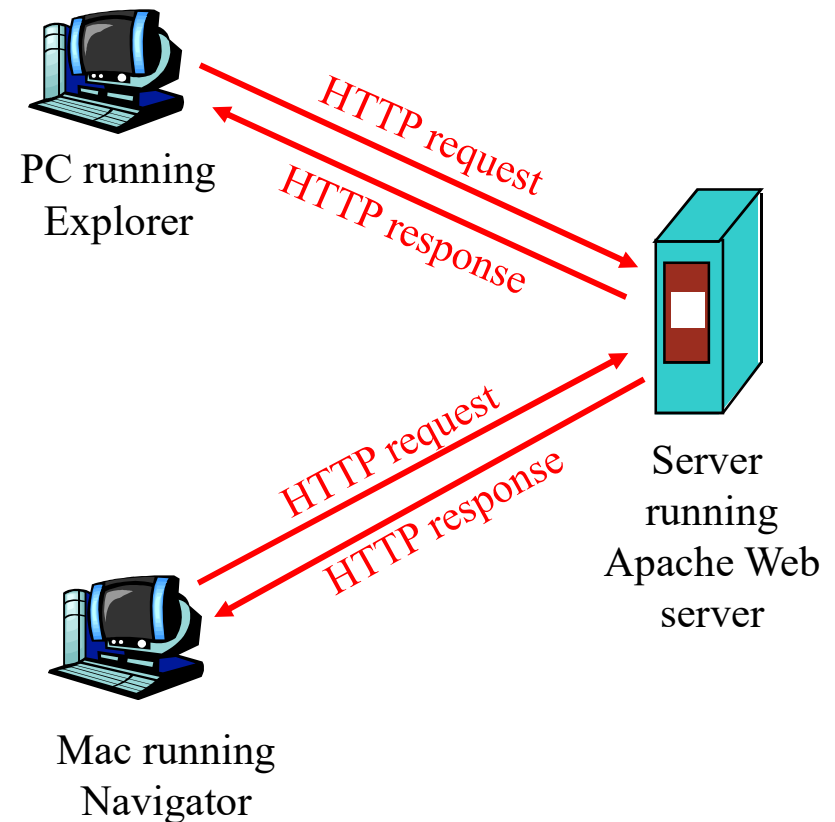
path name

# HTTP overview

## HTTP: hypertext transfer protocol

- Web's application layer protocol
- client/server model
  - *client*: browser that requests, receives, "displays" Web objects
  - *server*: Web server sends objects in response to requests

HTTP uses port no 80



# HTTP overview (continued)

## Uses TCP:

- client initiates TCP connection (creates socket) to server, port 80
- server accepts TCP connection from client
- HTTP messages (application-layer protocol messages) exchanged between browser (HTTP client) and Web server (HTTP server)
- TCP connection closed

## HTTP is “stateless”

- server maintains no information about past client requests

# HTTP connections

## Nonpersistent HTTP

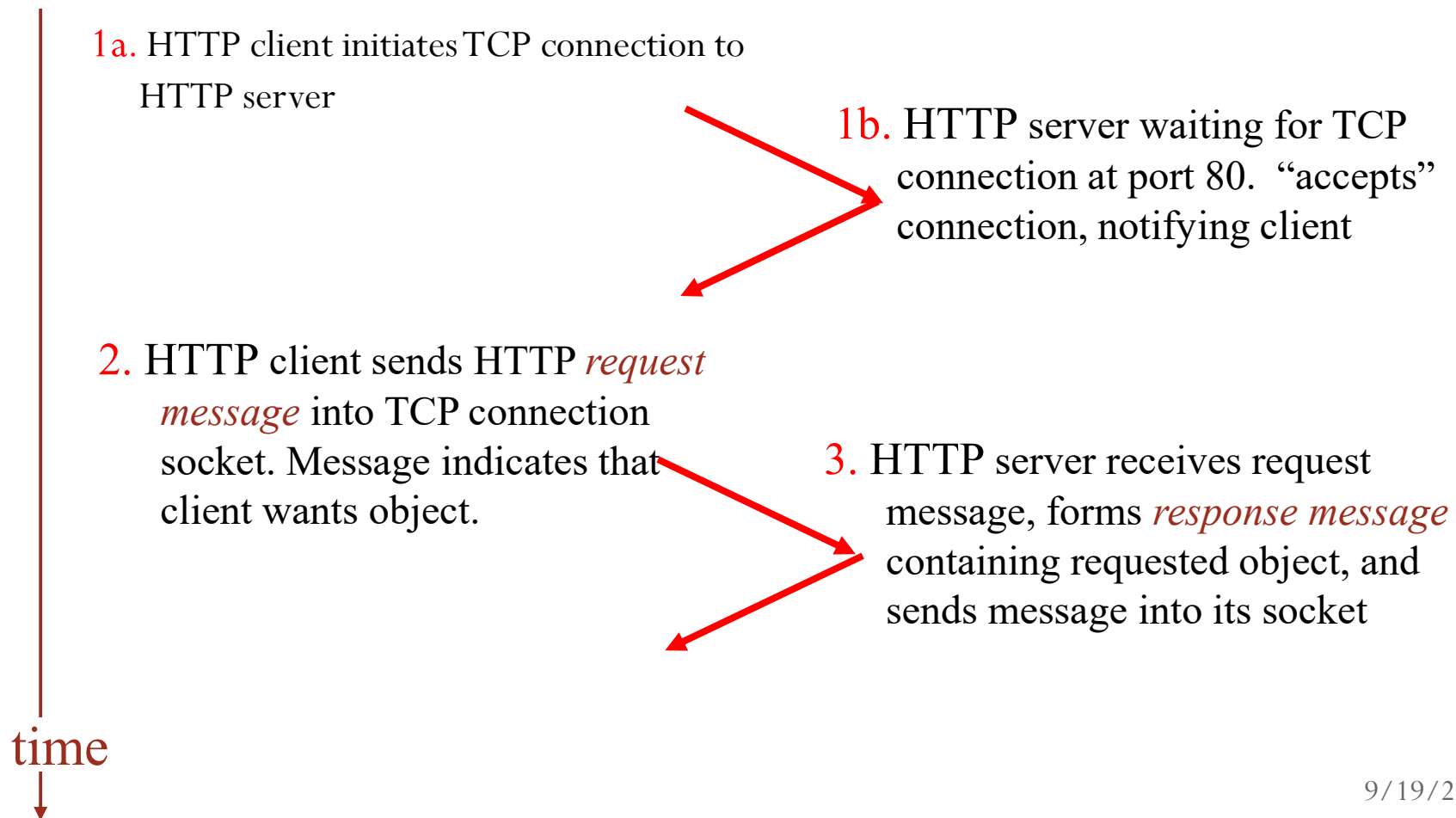
- At most one object is sent over a TCP connection.
- New connection must be established and maintained for each requested object
- requires 2 RTTs per object
- OS overhead for *each* TCP connection

## Persistent HTTP

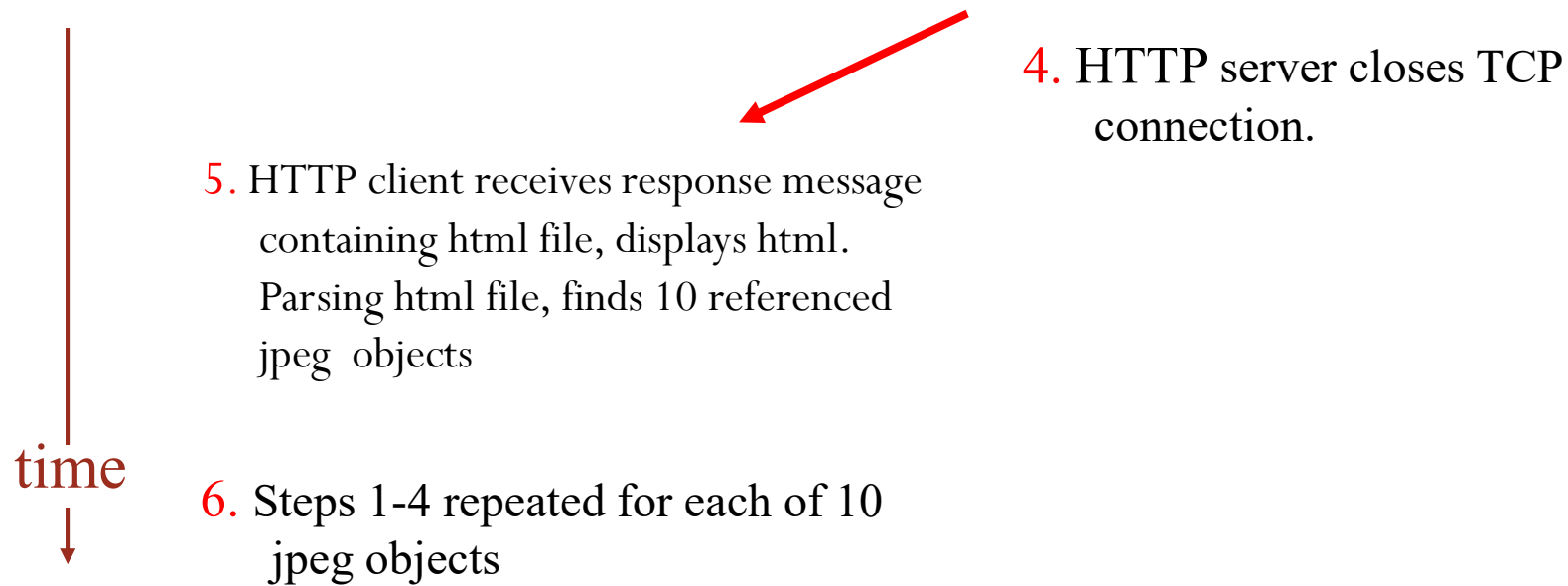
- Multiple objects can be sent over single TCP connection between client and server.
- server leaves connection open after sending response
- subsequent HTTP messages between same client/server sent over open connection
- client sends requests as soon as it encounters a referenced object
- as little as one RTT for all the referenced objects

# Nonpersistent HTTP

- Suppose user enters URL (contains text, references to 10 jpeg images)



# Nonpersistent HTTP (cont.)





# Non-Persistent HTTP: Response time

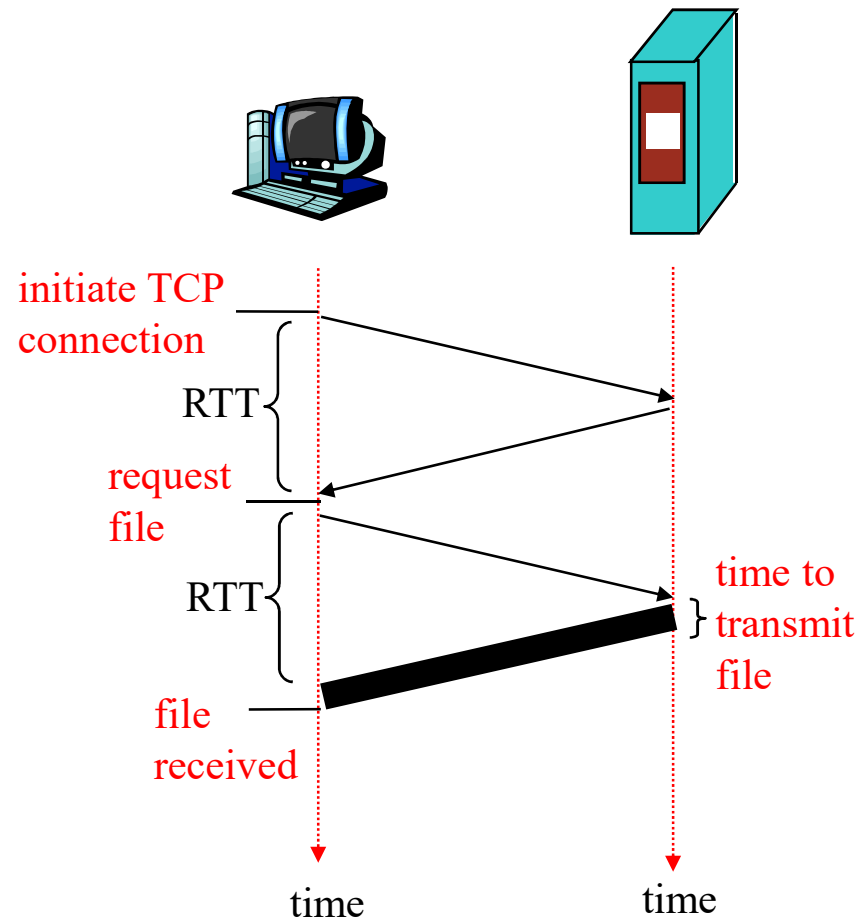
## Definition of RTT(Round Trip Time):

time for a small packet to travel from client to server and back.

## Response time:

- one RTT to initiate TCP connection
- one RTT for HTTP request
- file transmission time

**total = 2RTT + transmit time**



# HTTP Message Formats

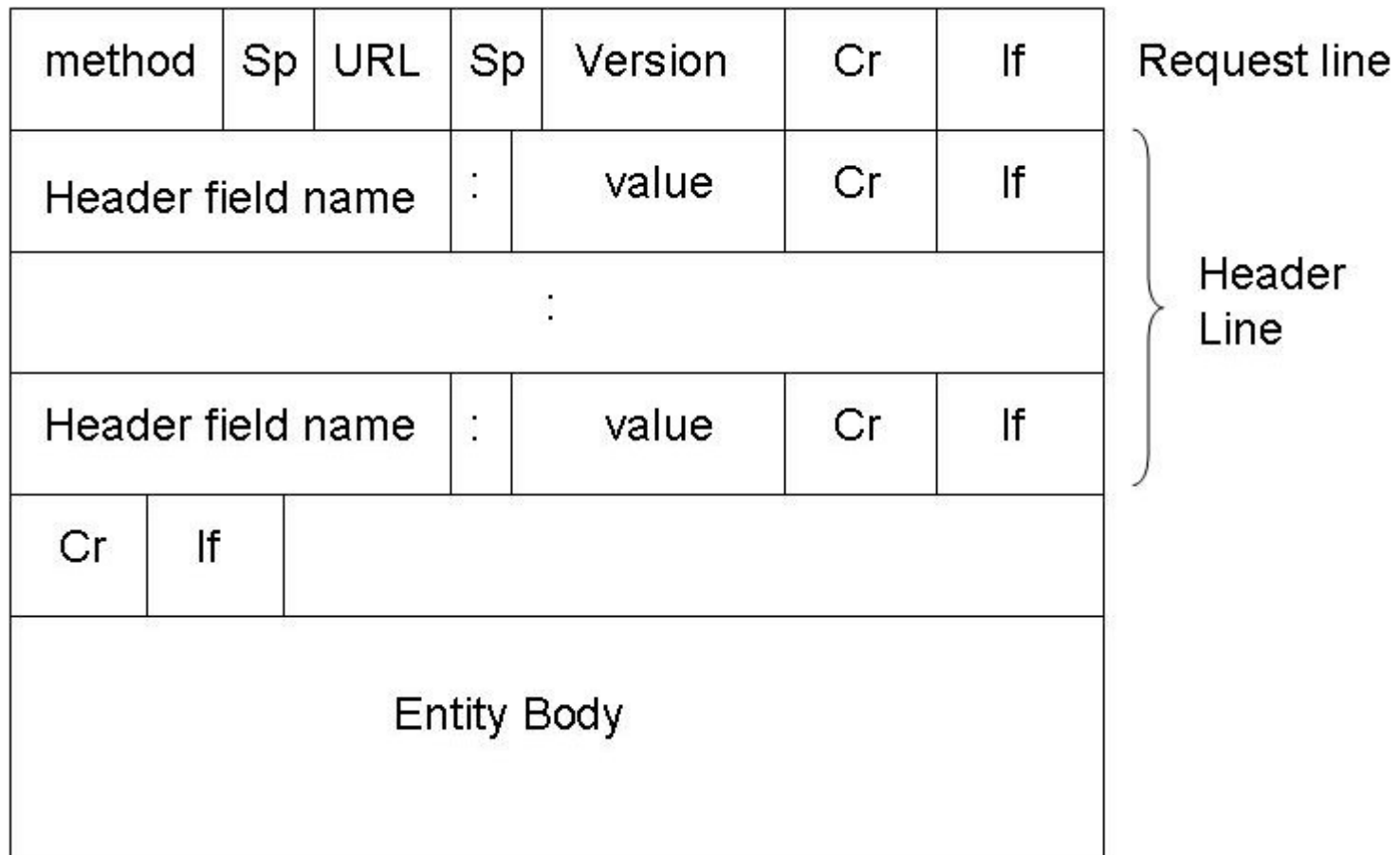
Two types of message

```
graph LR; A[Two types of message] --- B[HTTP Request message]; A --- C[HTTP Response Message]
```

HTTP  
Request  
message

HTTP  
Response  
Message

# HTTP request message



# HTTP request message Example

- HTTP request message:
  - ASCII (human-readable format)

request line  
(GET, POST,  
HEAD commands)

header  
lines

```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
User-agent: Mozilla/4.0
Connection: close
Accept-language: fr
```

Carriage return,  
line feed  
indicates end  
of message

(extra carriage return, line feed)

# Uploading form input

## Post method:

- Web page often includes form input
- Input is uploaded to server in entity body

## GET method:

- Uses GET method
- Input is uploaded in URL field of request line:

## **Example**

`www.somesite.com/animalsearch?monkeys&banana`

# HTTP Methods

Method	Description
GET	Request to read a Web page
HEAD	Request to read a Web page's header
PUT	Request to store a Web page
POST	Append to a named resource (e.g., a Web page)
DELETE	Remove the Web page
TRACE	Echo the incoming request
CONNECT	Reserved for future use
OPTIONS	Query certain options

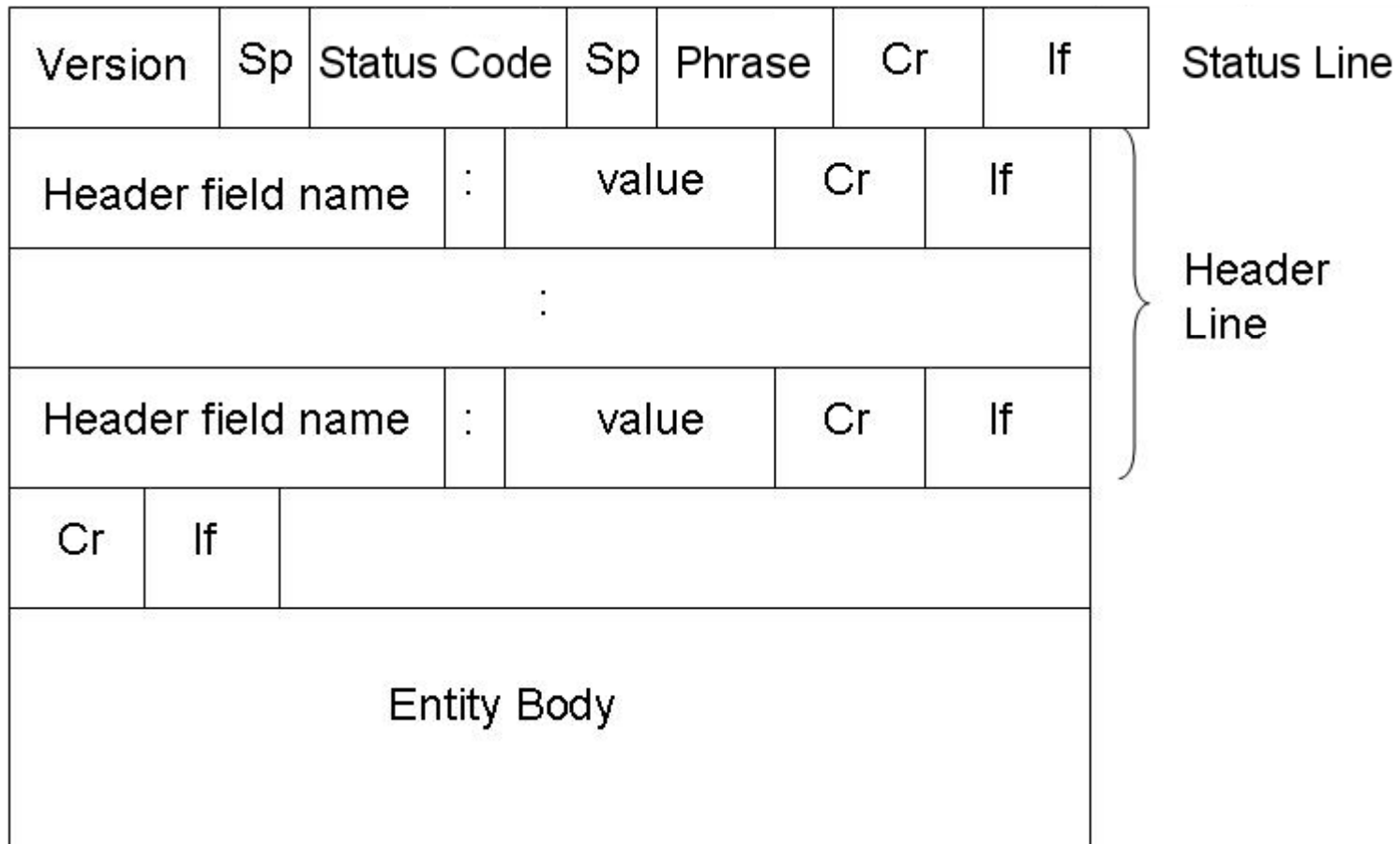
TRACE method is used for debugging. It instructs the server to send back the request. This is useful when requests are not processed correctly and the client wants to know it sent the proper request.

CONNECT – is not used at the moment

OPTIONS – provides a way for the client to query the server about its properties or those of a specified file

9/19/2017

# HTTP response message



# HTTP response message example

status line  
(protocol  
status code  
status phrase)

HTTP/1.1 200 OK

header  
lines

Connection close

Date: Thu, 06 Aug 1998 12:00:15 GMT

Server: Apache/1.3.0 (Unix)

Last-Modified: Mon, 22 Jun 1998 .....

Content-Length: 6821

Content-Type: text/html

data, e.g.,  
requested  
HTML file

data data data data data ...



# HTTP Version differences

## HTTP/1.0

Have only 16 status codes

Provides only basic authentication

Uses Non persistent connection

RTT is more so bandwidth waste is vast

Stateless

Supports only GET POST and HEAD method

## HTTP/1.1

Have introduced new 24 status code

Provides strong authentication

Uses Persistent connection

RTT is less so bandwidth utilization is good

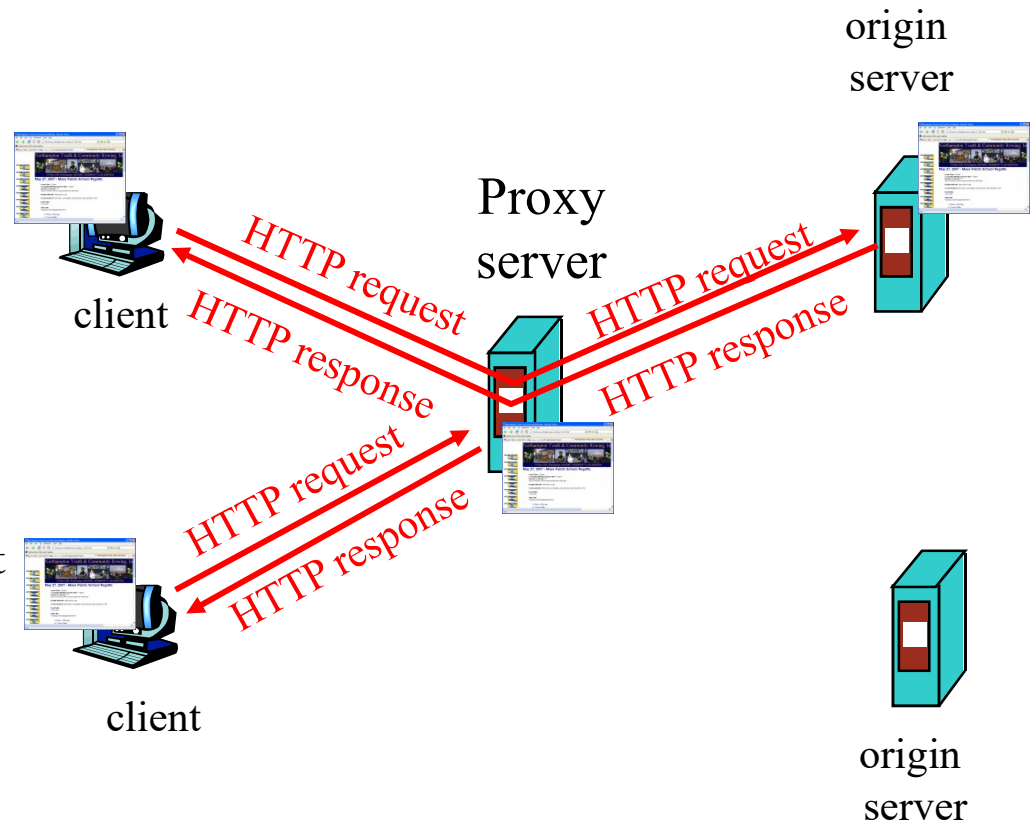
Uses cookies as state management mechanism

Supports GET, POST, HEAD,PUT and DELETE

# Web caches (proxy server)

**Goal:** satisfy client request without involving origin server

- user sets browser: Web accesses via cache
- browser sends all HTTP requests to cache
  - object in cache: cache returns object
  - else cache requests object from origin server, then returns object to client



# Outline

Hyper Text Transfer Protocol (HTTP),

**FTP,**

Email: SMTP, MIME, POP3, Webmail,

Domain Name System (DNS),

TELNET,

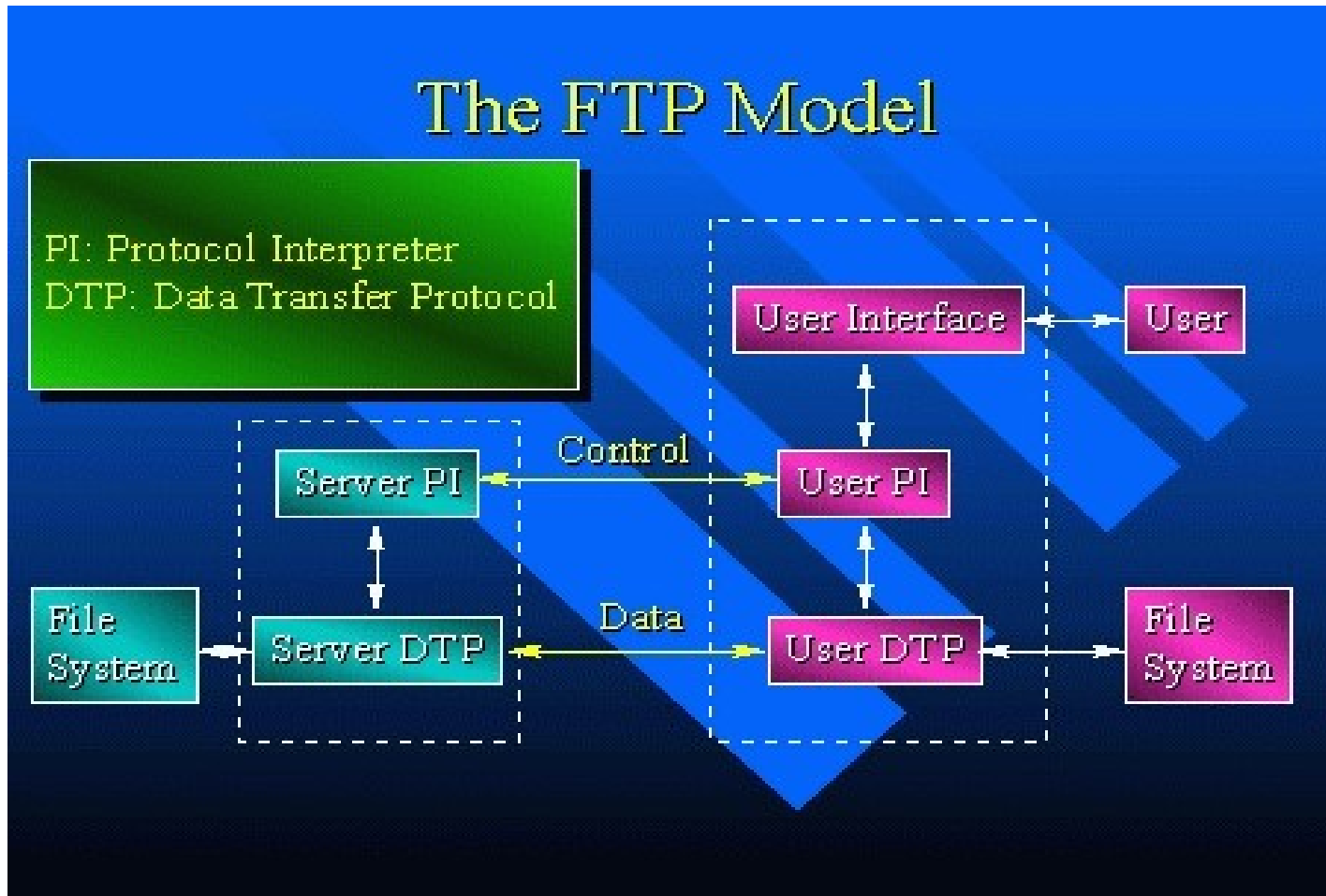
Dynamic Host Control Protocol (DHCP),

Simple Network Management Protocol (SNMP).

# FTP (File Transfer Protocol)

- **Objective**
  1. Allow file sharing between remote machine
  2. Transfer data reliably and efficiently
- FTP Protocol falls within **client server model**
- Both client & server have **2 process** allowing information (Data & command) to be managed , they are
  1. DTP(Data transfer Process)
  2. PI(Protocol Interpreter)

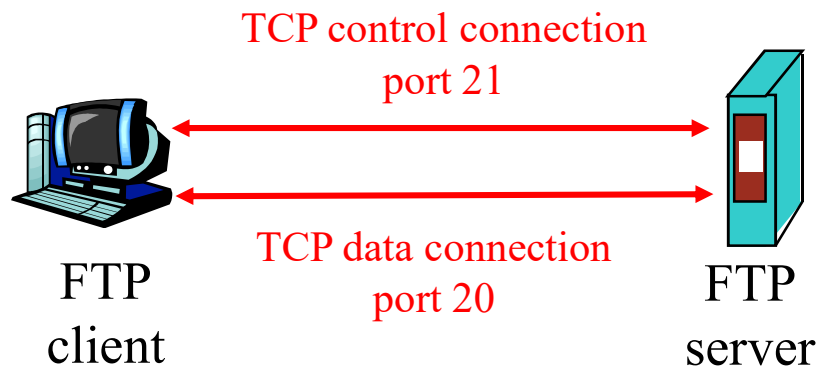
# FTP model/connection



# FTP: the file transfer protocol

- During FTP Connection 2 transmission channels are open
  1. Control Channel (For command)
  2. Data Channel (For data)
- Control uses port no 21
- Data connection uses Port no 20

# FTP: separate control, data connections



- FTP client contacts FTP server at port 21
- client authorized over control connection
- client browses remote directory by sending commands over control connection.
- when server receives file transfer command, server opens 2<sup>nd</sup> TCP connection (for file) to client
- after transferring one file, server closes data connection.
- FTP server maintains “state”: current directory, earlier authentication

# FTP transmission modes

- **Stream mode:** Data is sent as a **continuous stream**, relieving FTP from doing any processing. Rather, all processing is left up to TCP. No End-of-file indicator is needed, unless the data is divided into records.
- **Block mode:** FTP **breaks the data into several blocks** (block header, byte count, and data field) and then passes it on to TCP.
- **Compressed mode:** **Data is compressed** using a simple algorithm (usually run-length encoding).



# FTP commands, responses

## Sample commands:

- sent as ASCII text over control channel
- **USER** *username*
- **PASS** *password*
- **LIST** return list of file in current directory
- **RETR filename** retrieves (gets) file
- **STOR filename** stores (puts) file onto remote host

## Sample return codes

- status code and phrase (as in HTTP)
- **331 Username OK, password required**
- **125 data connection already open; transfer starting**
- **425 Can't open data connection**
- **452 Error writing file**

# Comparison between HTTP and FTP

- HTTP uses TCP
- Only one TCP connection between client and server
- Send its control information in-band
- HTTP is stateless
- Port No is 80
- FTP uses TCP
- Two parallel TCP connection, one for data and one for control
- Send its control information out-of-band
- FTP maintains State
- Port no is 20 and 21

Question Time!

What is meant by  
Anonymous FTP?

# Outline

Hyper Text Transfer Protocol (HTTP),

FTP,

**Email: SMTP, MIME, POP3, Webmail,**

Domain Name System (DNS),

TELNET,

Dynamic Host Control Protocol (DHCP),

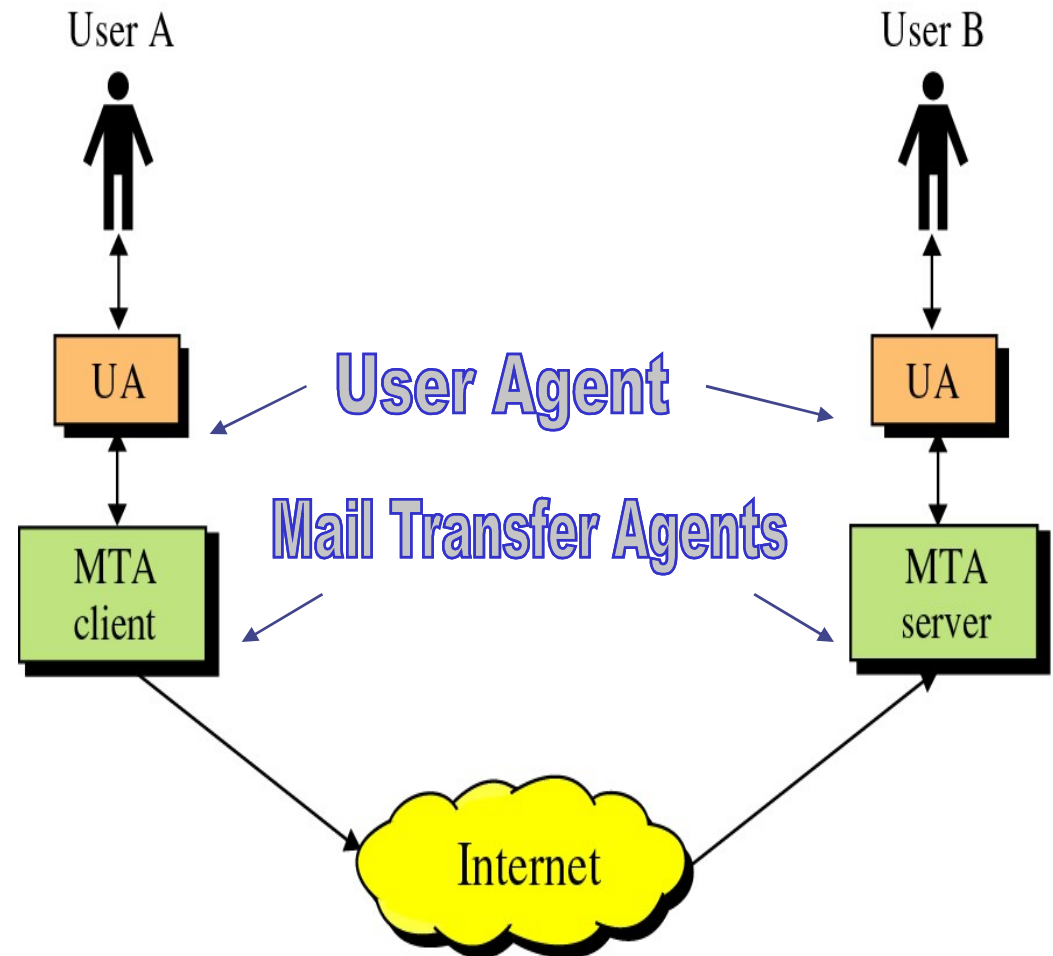
Simple Network Management Protocol (SNMP).

# What is SMTP?

- The standard for e-mail transmissions across the Internet
- It is defined in RFC 821
- It is a relatively simple, text-based protocol
- Port No is 25
- It uses reliable connection to send mail. For the same it uses TCP connection.

# SMTP

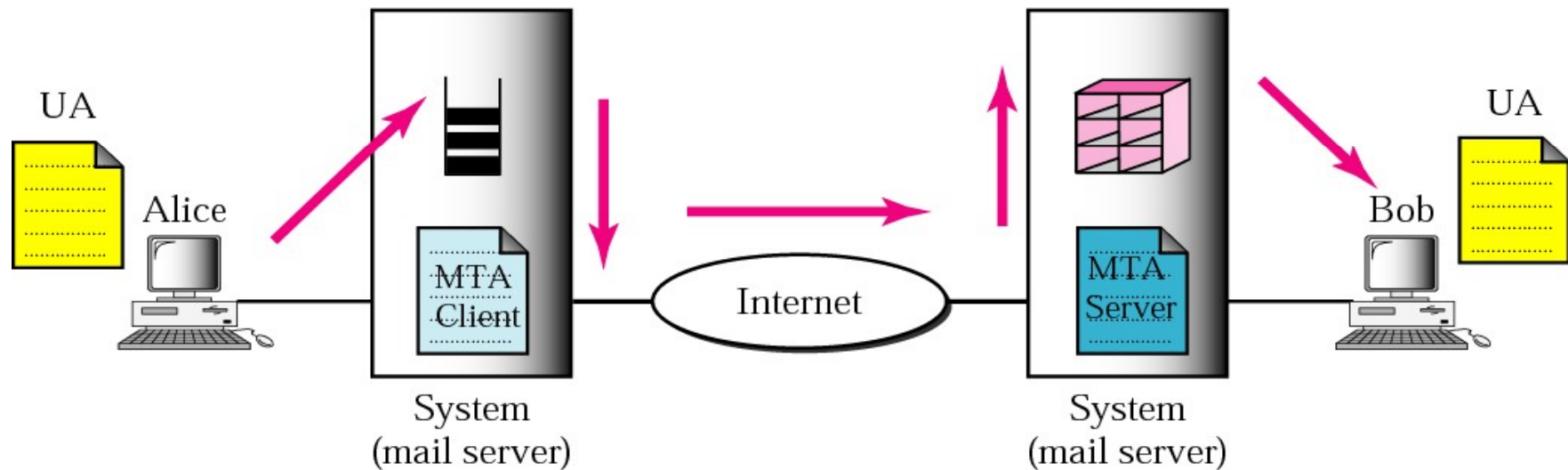
- SMTP clients and servers have two main components
  - **User Agents** – Prepares the message, encloses it in an envelope.
  - **Mail Transfer Agent (MTA)** – Transfers the mail across the internet



# SMTP Working with UA and MTA

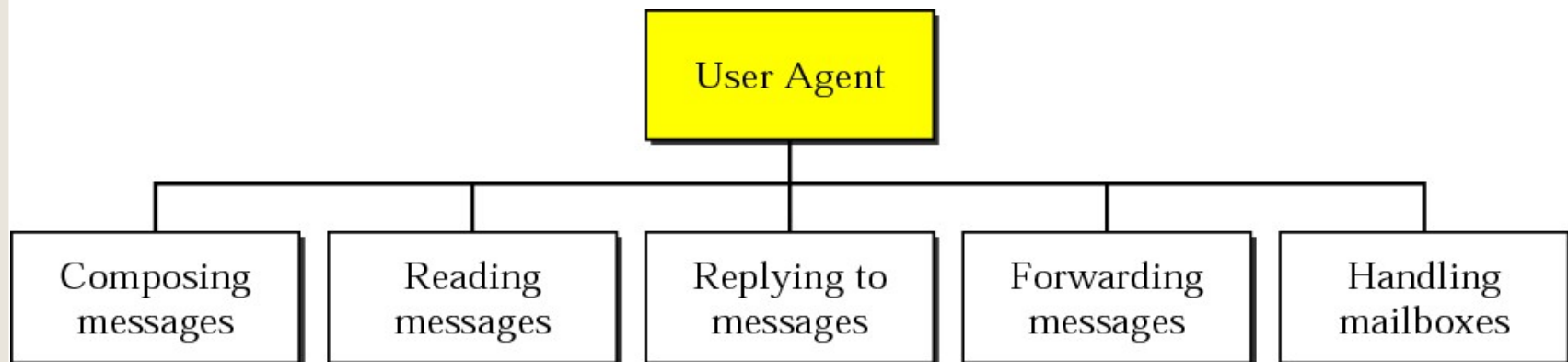
UA: user agent

MTA: message transfer agent



# USER AGENT

*The user agent (UA) provides service to the user to make the process of sending and receiving a message easier.*





# What is Mail?

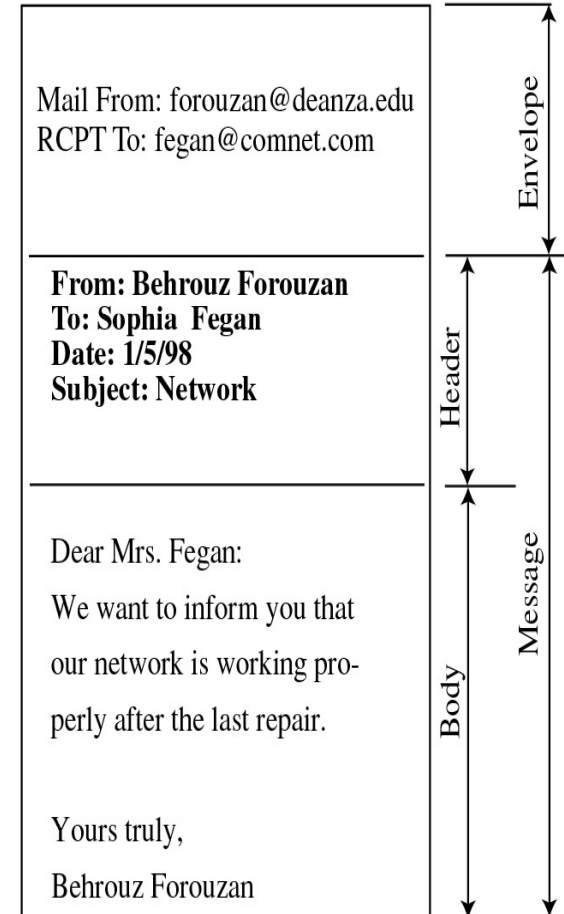
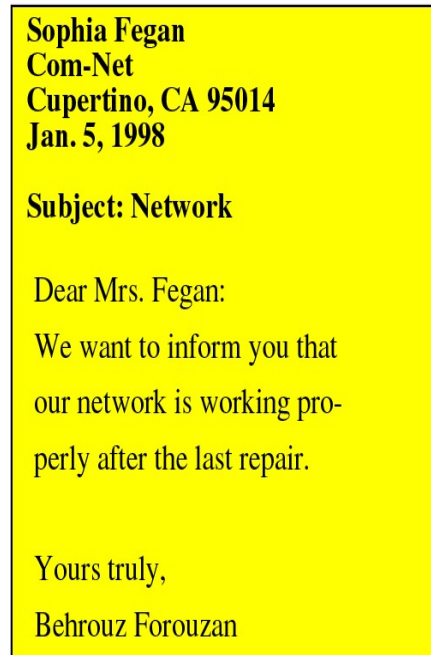
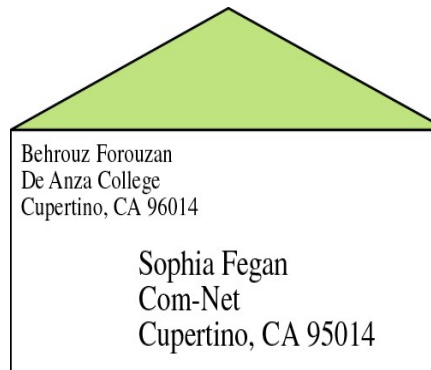
◆ Mail is a text file

◆ Envelope –

- sender address
- receiver address
- other information

◆ Message –

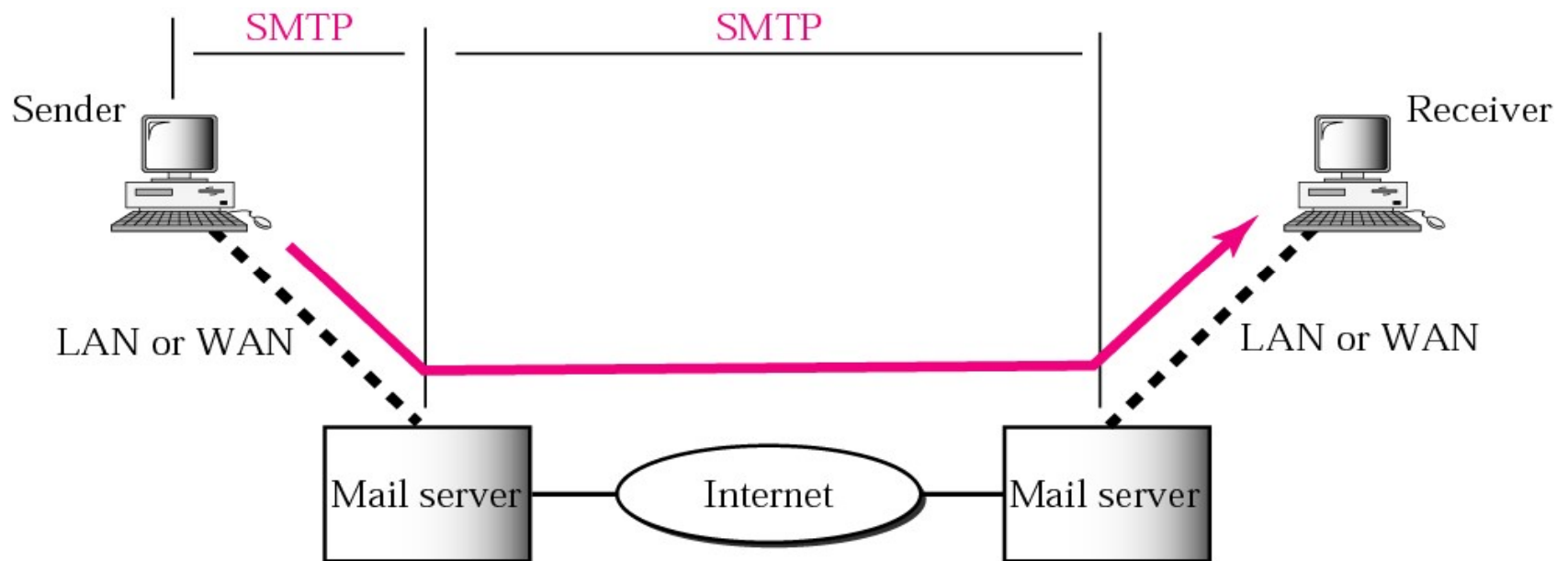
- Mail Header – defines the sender, the receiver, the subject of the message, and some other information
- Mail Body – Contains the actual information in the message



# MESSAGE TRANSFER AGENT: SMTP

*The actual mail transfer requires message transfer agents (MTAs). The protocol that defines the MTA client and server in the Internet is called Simple Mail Transfer Protocol (SMTP).*

Figure SMTP range



*Table 20.4 Commands*

<i>Keyword</i>	<i>Argument(s)</i>
HELO	Sender's host name
MAIL FROM	Sender of the message
RCPT TO	Intended recipient of the message
DATA	Body of the mail
QUIT	
RSET	
VERFY	Name of recipient to be verified
NOOP	
TURN	
EXPN	Mailing list to be expanded
HELP	Command name
SEND FROM	Intended recipient of the message
SMOL FROM	Intended recipient of the message
SMAL FROM	Intended recipient of the message

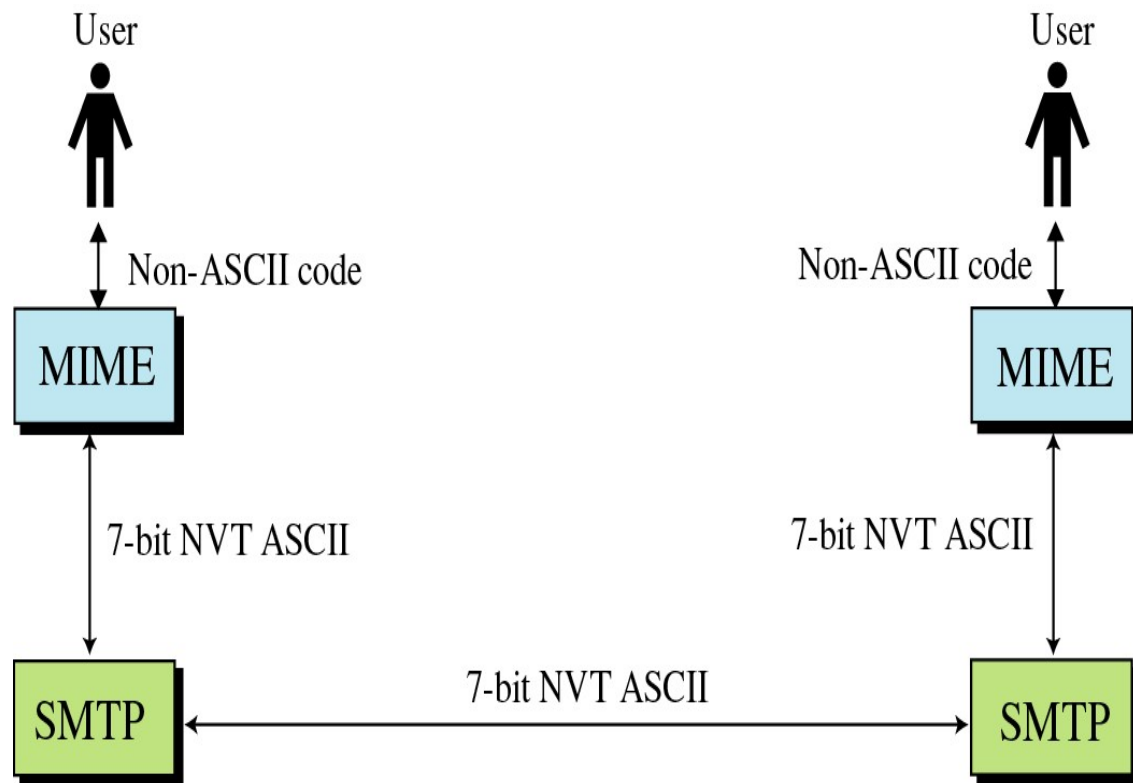
# Status Codes

- ◆ The Server responds with a 3 digit code that may be followed by text info
  - 2### - Success
  - 3### - Command can be accepted with more information
  - 4### - Command was rejected, but error condition is temporary
  - 5### - Command rejected, Bad User!

# Extensions to SMTP

## ◆ MIME – Multipurpose Internet Mail Extensions

- Transforms non-ASCII data to NVT (Network Virtual Terminal) ASCII data
  - ◆ Text
  - ◆ Application
  - ◆ Image
  - ◆ Audio
  - ◆ Video



# MIME and Base64 Encoding

- ◆ If the internet is the information highway, then the path for email is a narrow tunnel
  - Only very small vehicles can pass through
- ◆ Then how do you send a big truck through a small ravine?
  - You have to break it down to smaller pieces and transport the pieces through the ravine, and reassemble the truck

# MIME and Base64 Encoding

- ◆ The same happens when you send a file attachment via email.
- ◆ This is known as encoding
  - the binary data (256 bits) is transformed to ASCII text (128 bits)
  - allowing it to fit through the tunnel
- ◆ On the recipient's end, the data is decoded and the original file is rebuilt.



# MESSAGE ACCESS AGENT: POP AND IMAP

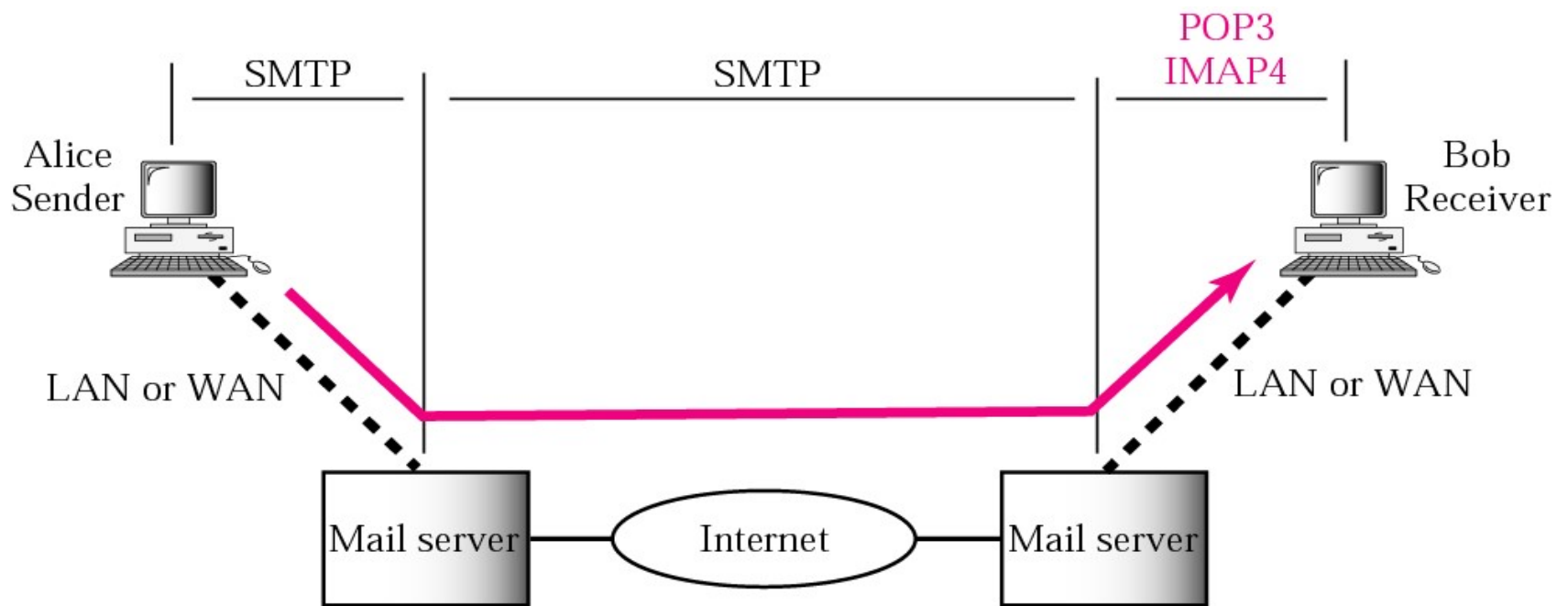
*The third stage of mail delivery uses a message access agent; the client must pull messages from the server. Currently two message access protocols are available: Post Office Protocol, version 3 (POP3) and Internet Mail Access Protocol, version 4.*

*The topics discussed in this section include:*

*POP3*

*IMAP4*

Figure POP3 and IMAP4



# POP3

- Starts when the user starts the mail reader
- Mail reader calls up the ISP (if there is no connection) and establishes a TCP connection with the message transfer agent on port 110;
  - Authorization
    - Having user logged in by sending its username and password
  - Transactions
    - User collecting the e-mails and marking them for deletion
  - Update
    - Causes the e-mails to be deleted

# IMAP

- POP3 works fine for users with one e-mail account with one ISP, accessed from one PC
  - If mail was accessed from different locations, user may lose e-mails, security issues may appear, etc
- An alternative mail delivery protocol, IMAP (Internet Message Access Protocol), defined in RFC2060
  - Instead of assuming that all messages will be downloaded and work offline after that (like POP3), IMAP assumes that all e-mail will remain on the server indefinitely in multiple mailboxes
  - Provides extensive mechanisms to read messages or parts of messages, mechanisms to create, destroy and manipulate multiple mailboxes.

# POP3 vs. IMAP

<b>Feature</b>	<b>POP3</b>	<b>IMAP</b>
Where is protocol defined?	RFC 1939	RFC 2060
Which TCP port is used?	110	143
Where is e-mail stored?	User's PC	Server
Where is e-mail read?	Off-line	On-line
Connect time required?	Little	Much
Use of server resources?	Minimal	Extensive
Multiple mailboxes?	No	Yes
Who backs up mailboxes?	User	ISP
Good for mobile users?	No	Yes
User control over downloading?	Little	Great
Partial message downloads?	No	Yes
Are disk quotas a problem?	No	Could be in time
Simple to implement?	Yes	No
Widespread support?	Yes	Growing

# Outline

Hyper Text Transfer Protocol (HTTP),

FTP,

Email: SMTP, MIME, POP3, Webmail,

**Domain Name System (DNS),**

TELNET,

Dynamic Host Control Protocol (DHCP),

Simple Network Management Protocol (SNMP).

# DNS (Domain Name System) Basic

IP addresses are difficult to remember

Need some mechanisms to translate gmail.com to the IP address.

To solve this problem DNS was invented

DNS is a hierarchical, domain-based naming scheme and a distributed database system for implementing this naming scheme

Usage: Map URL name into an IP address

# DNS: Domain Name System Basic:

Maps URL to IP address and vice versa

Runs over UDP and TCP

Uses port No 53

Application layer protocol



# DNS : DNS services

hostname to IP address translation

host aliasing

- Canonical, alias names

mail server aliasing

load distribution

- replicated Web servers: set of IP addresses for one canonical name

# DNS components

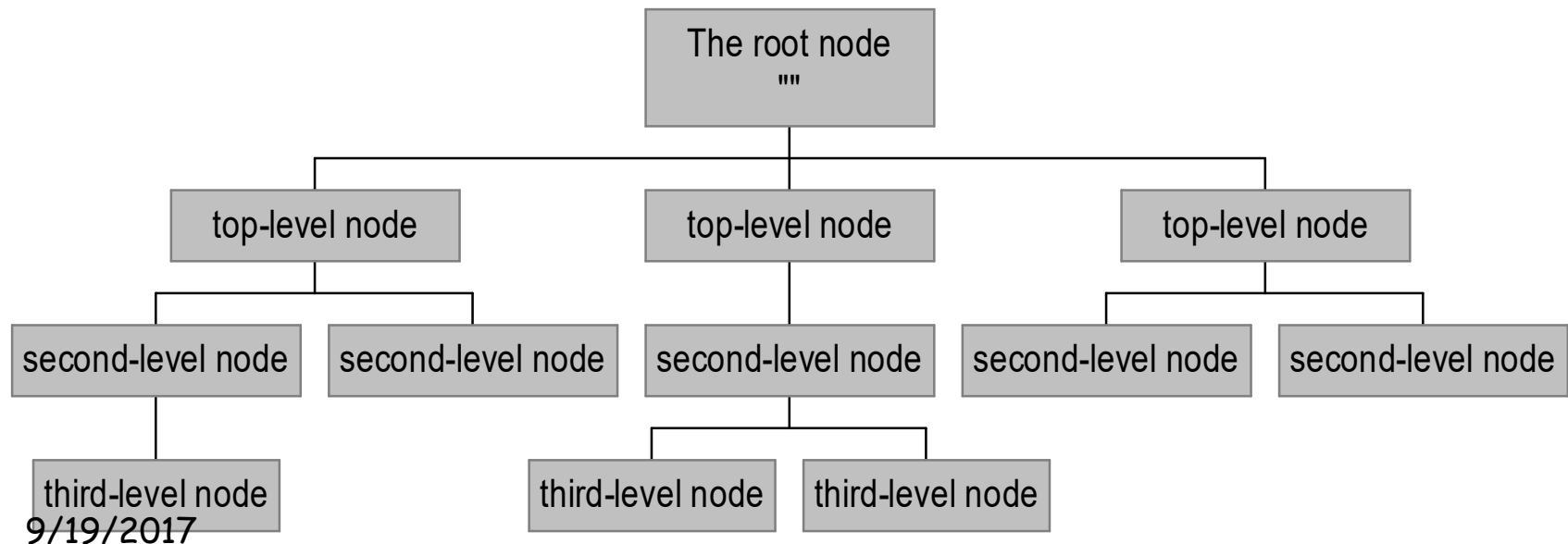
Name space

Name Servers

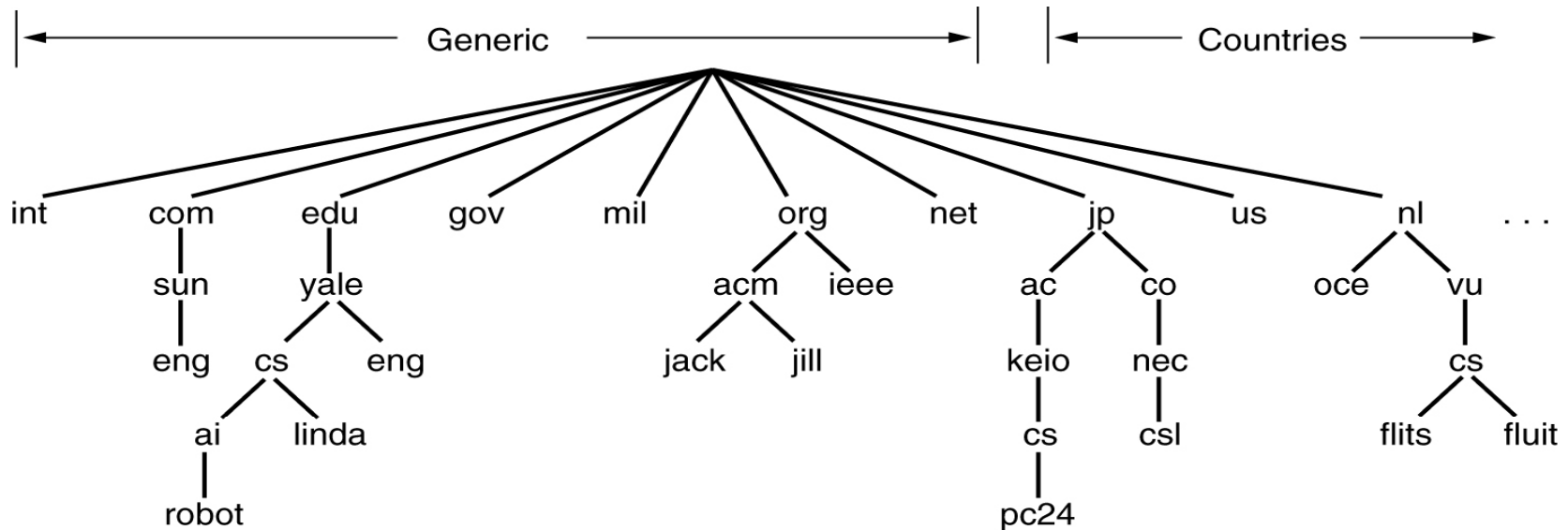
Resolver

# The Name Space

- ❑ The *name space* is the structure of the DNS database
  - ❖ An inverted tree with the root node at the top
- ❑ Each node has a label
  - ❖ The root node has a null label, written as ""



# DNS Name Space (Top level Domain)



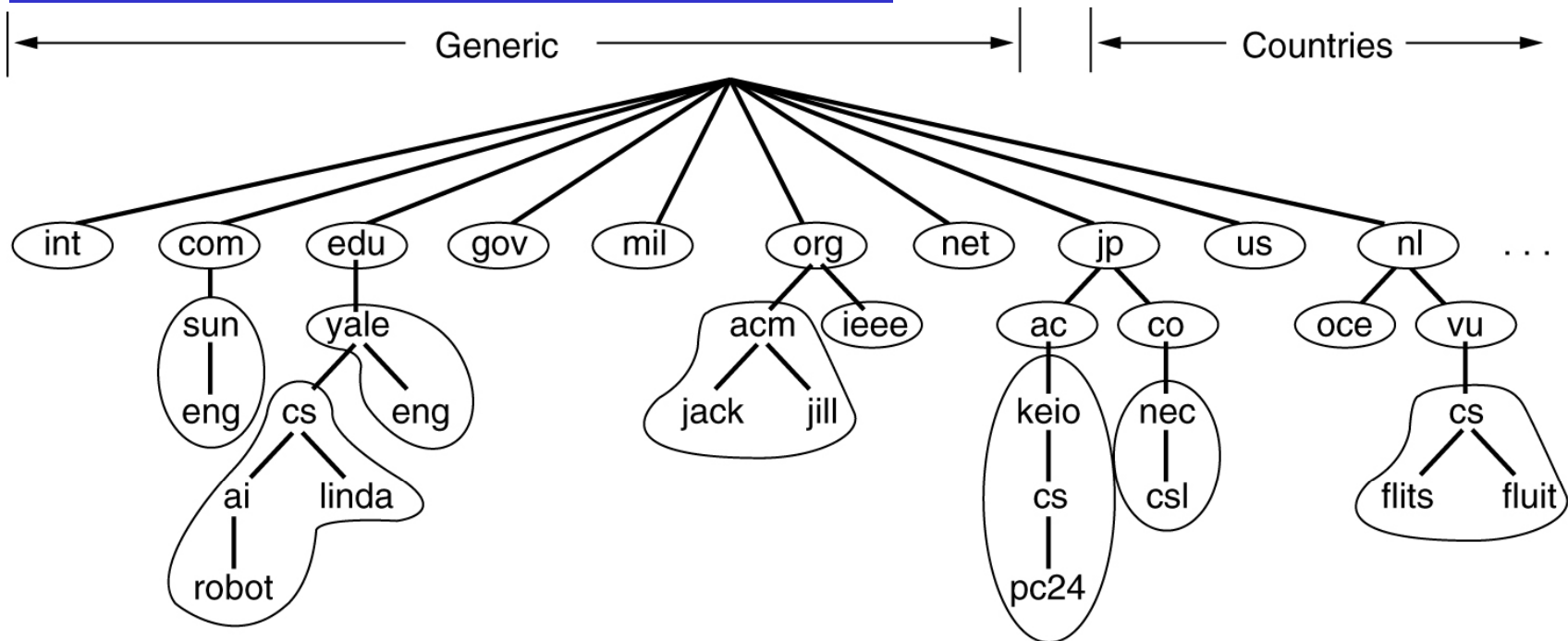
Internet is divided into over 200 top level domains

Top level domains could be **generic and country domains**

Each domain is divided into sub-domains, which are further partitioned, etc..

9/19/2017

# DNS Name servers



One DNS server could service all requests

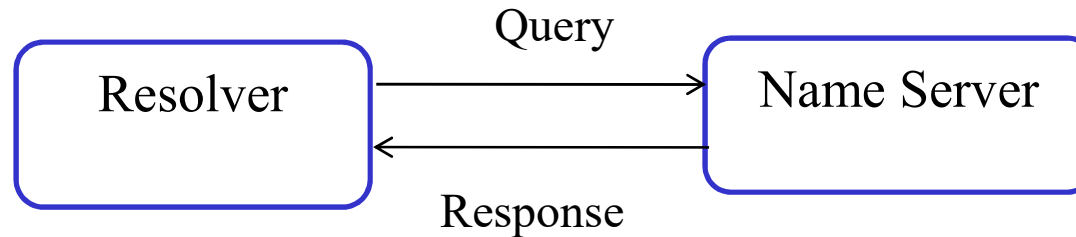
**But in practice it will be overloaded**

9/19/2017

To solve this, DNS name space is divided in non overlapping zones

**Each zone contains some part of the tree and name servers holding zone info**

# DNS Name Resolver



Resolver maps a name to an IP address and vice versa

There are 3 classes of DNS servers

- Root DNS Server
- Top-Level Domain(TLD) Server
- Authoritative DNS Server

# DNS Resolver (Types of Servers)

## Root DNS Server

- 13 root DNS server
- Labeled A to M

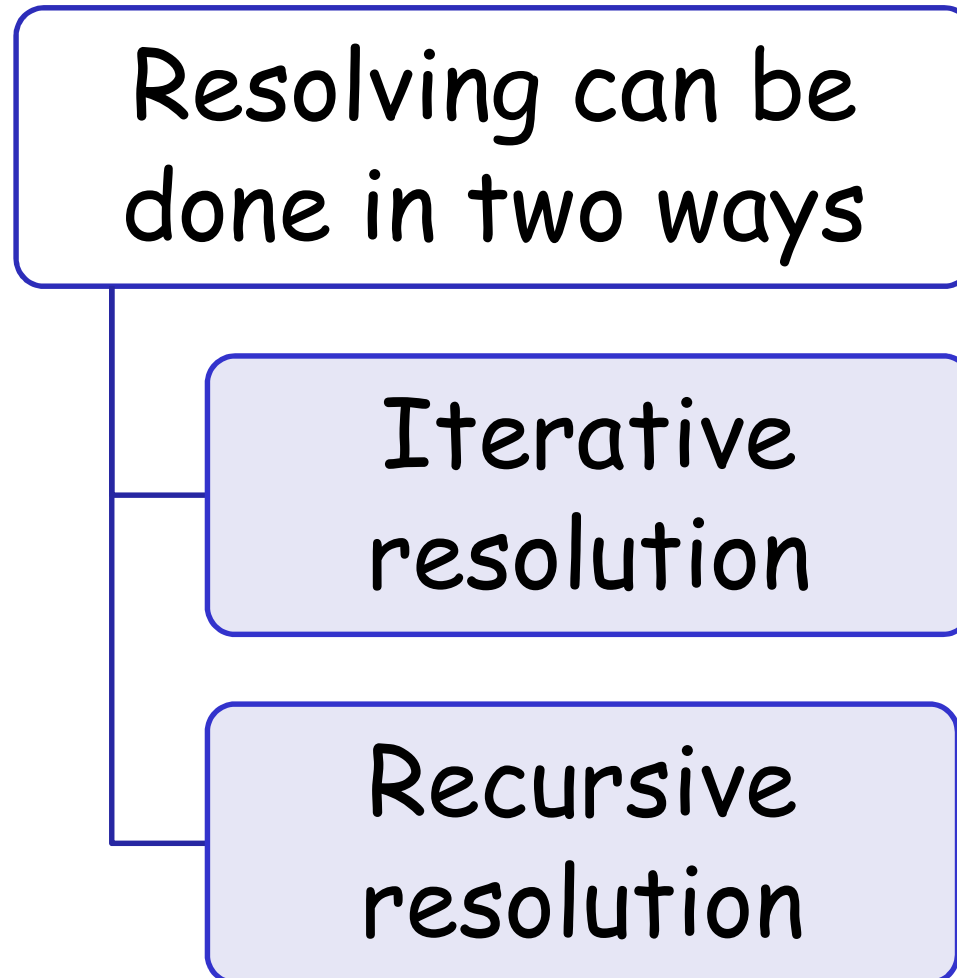
## Top-Level Domain(TLD) Servers

- Responsible for top level domains
- Ex. Com,org,edu,gov and all country domains

## Authoritative DNS Servers

- Contains DNS record for every publically accessible organization on Internet
- DNS record maps names to IP address

# DNS Resolver ways



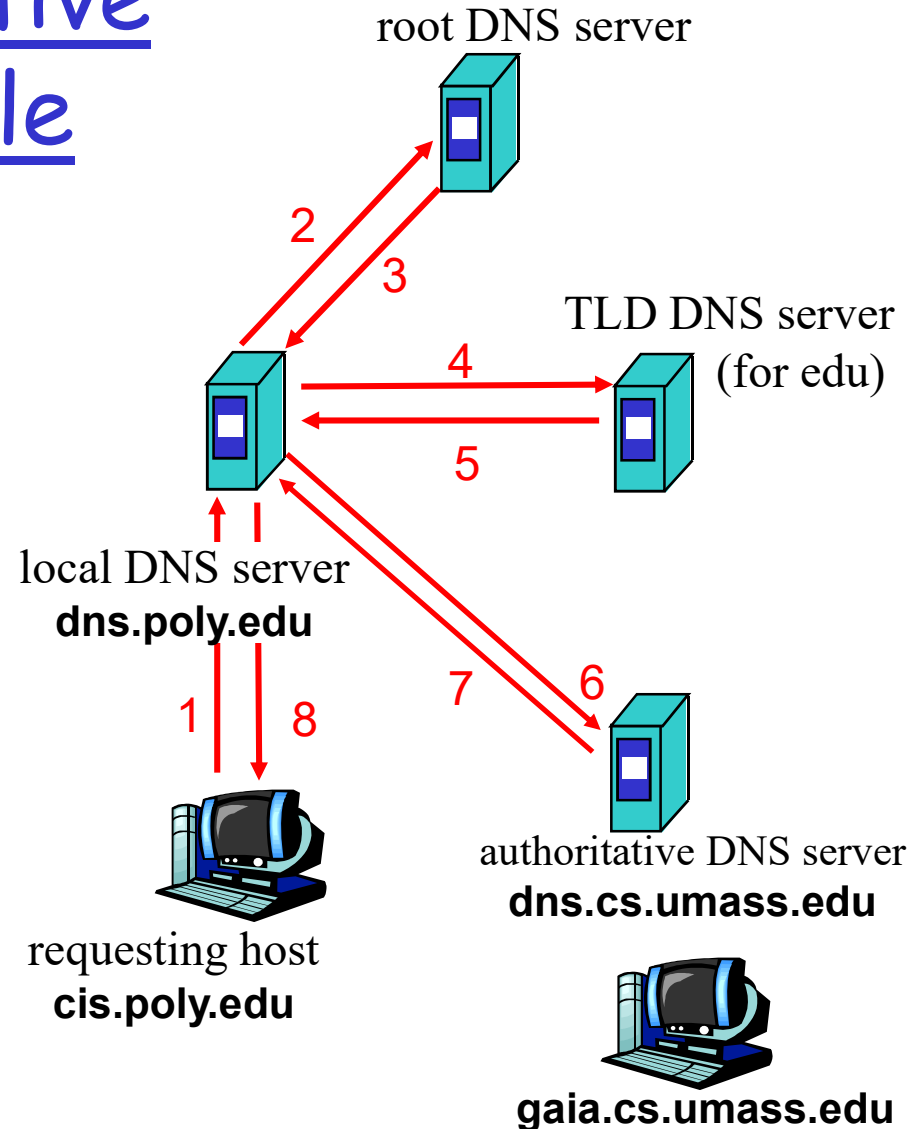


# DNS name Iterative resolution example

- Host at cis.poly.edu wants IP address for gaia.cs.umass.edu

## iterated query:

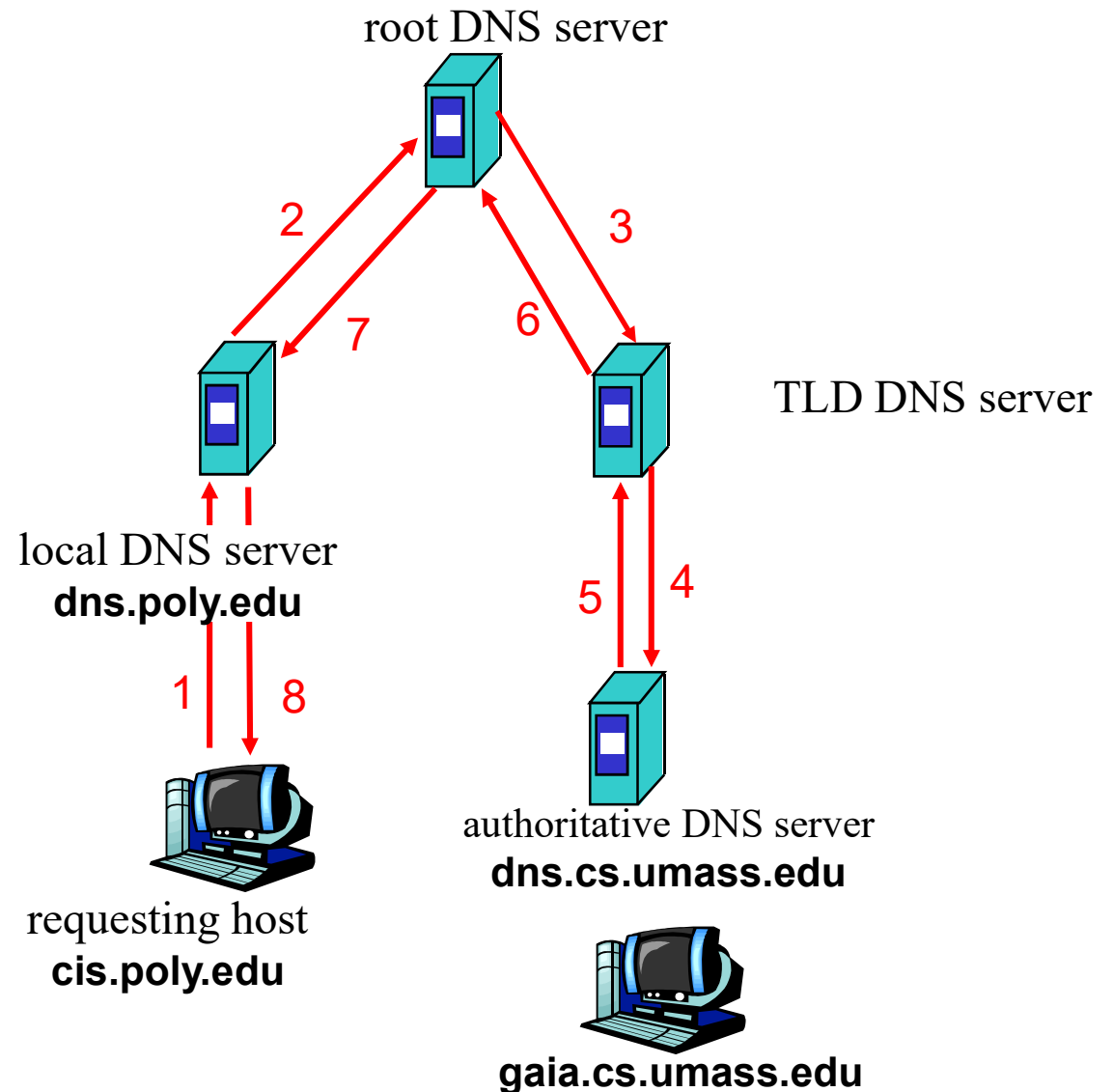
- contacted server replies with name of server to contact
- “I don’t know this name, but ask this server”



# DNS name Recursive resolution example

## recursive query:

- puts burden of name resolution on contacted name server
- heavy load?



# Resource Records

- ❑ When a resolver gives a domain name to DNS, what it gets back are the resource records associated with that name;
- ❑ Thus the primary function of DNS is to map domain names onto resource records
- ❑ Types of Resource Record are as given in the table.

<b>A</b>	Host Addresses
<b>PTR</b>	Reverse address name mapping
<b>CNAME</b>	Aliases
<b>MX</b>	Mail exchange for the domain
<b>NS</b>	Authoritative Name Servers

9/19/2017

# Parts Resource Records

Domain name

Time to Live

Class

Type

Value

9/19/2017

# Outline

Hyper Text Transfer Protocol (HTTP),

FTP,

Email: SMTP, MIME, POP3, Webmail,

Domain Name System (DNS),

**TELNET,**

Dynamic Host Control Protocol (DHCP),

Simple Network Management Protocol (SNMP).

# TELNET vs. telnet

- TELNET is a *protocol* that provides “a general, bi-directional, eight-bit byte oriented communications facility”.
- **telnet** is a *program* that supports the TELNET protocol over TCP.
- Many application protocols are built upon the TELNET protocol.

# TELNET Protocol

*TELNET is a general-purpose client-server application program.*

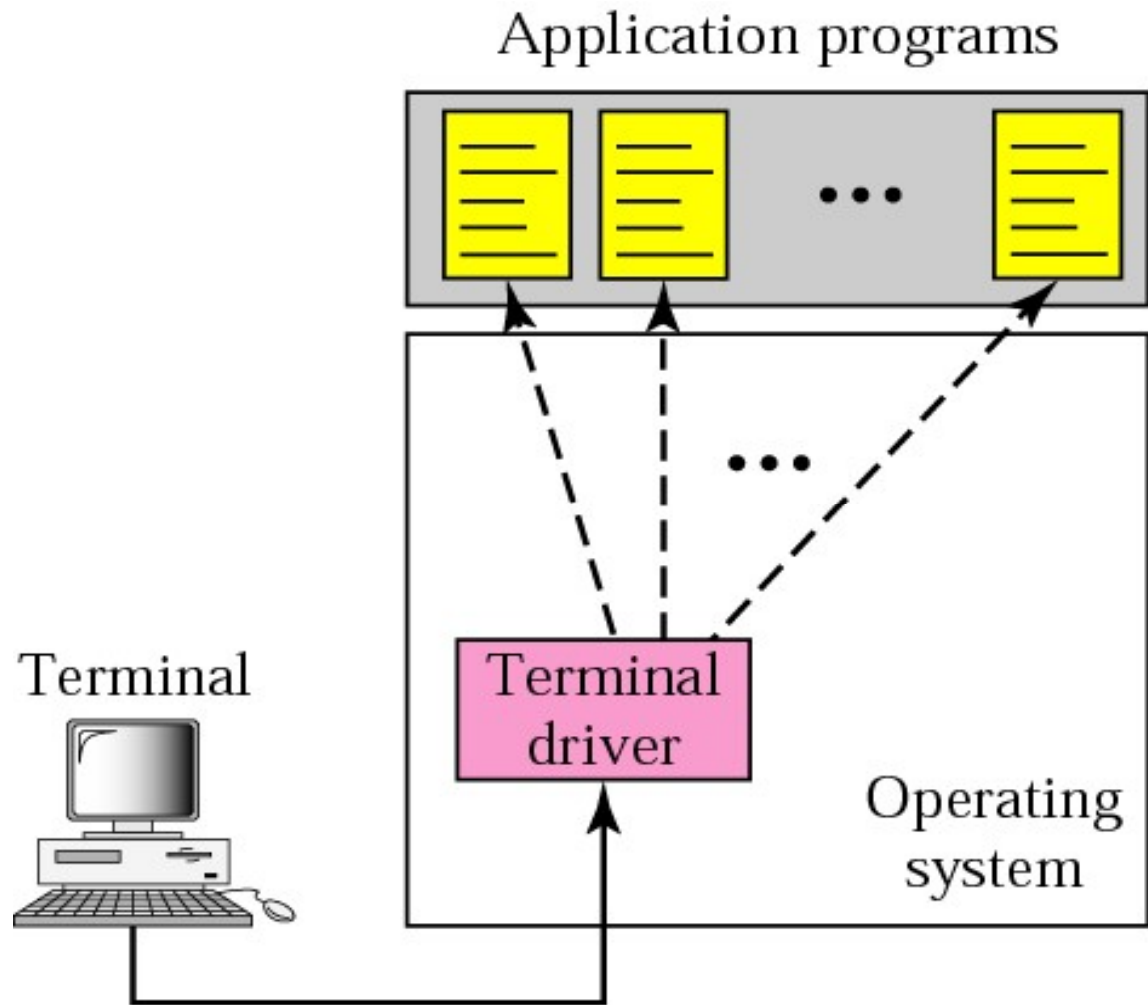
*TELNET is an abbreviation for TErminaL NETwork*

*Uses Port No 23 and TCP Protocol*

*Used for Virtual terminal service*

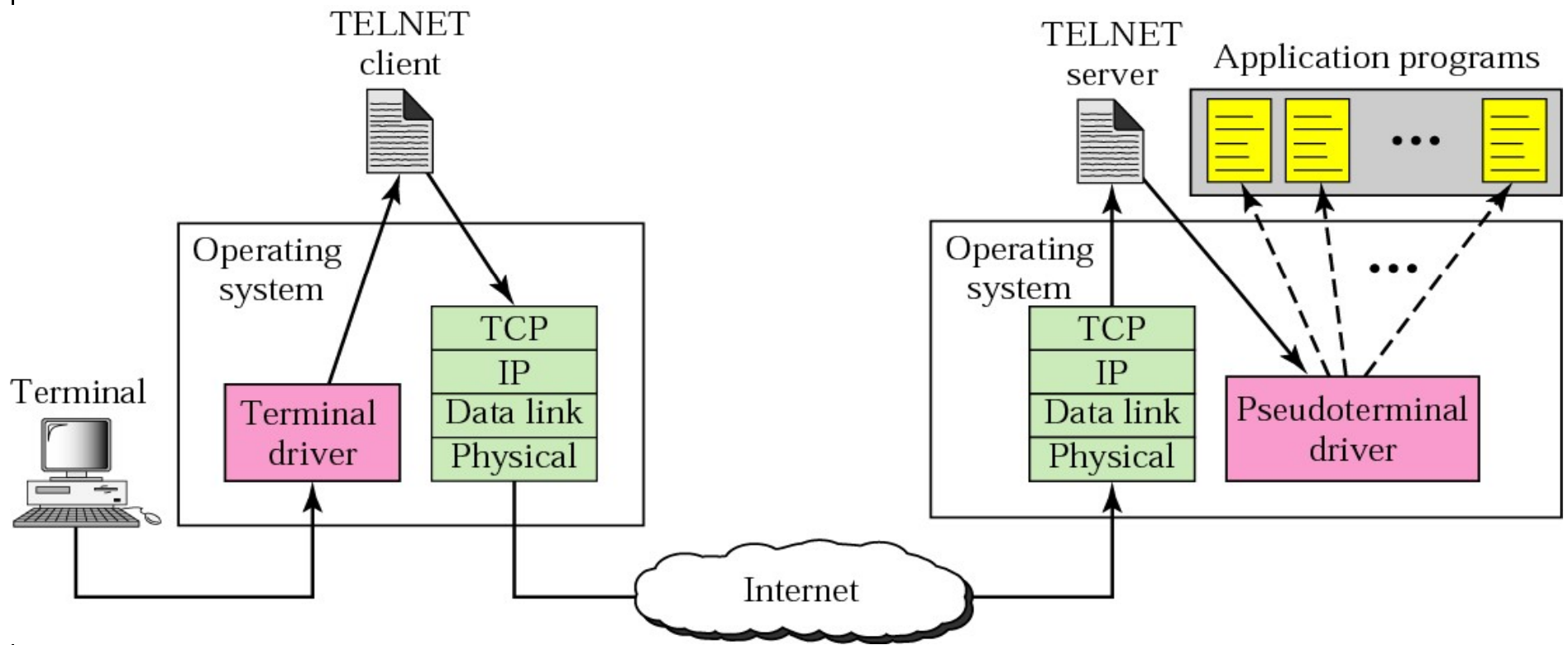
*TELNET enables the establishment of a connection to a remote system in such a way that the local terminal appears to be a terminal at the remote system.*

# Local login

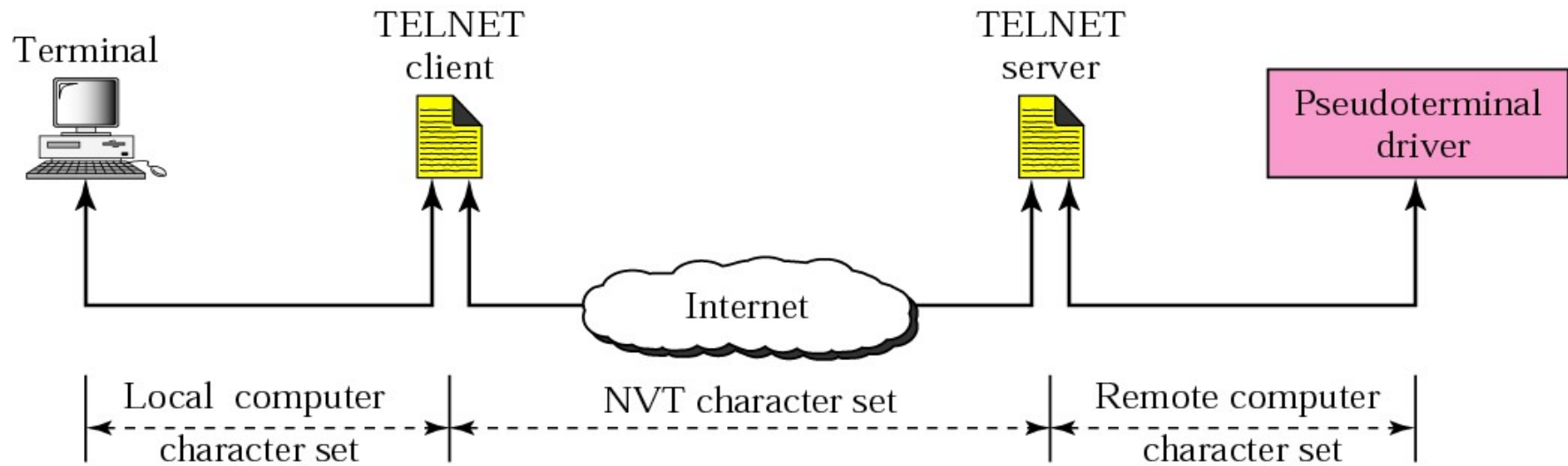




# Remote login



# Concept of NVT



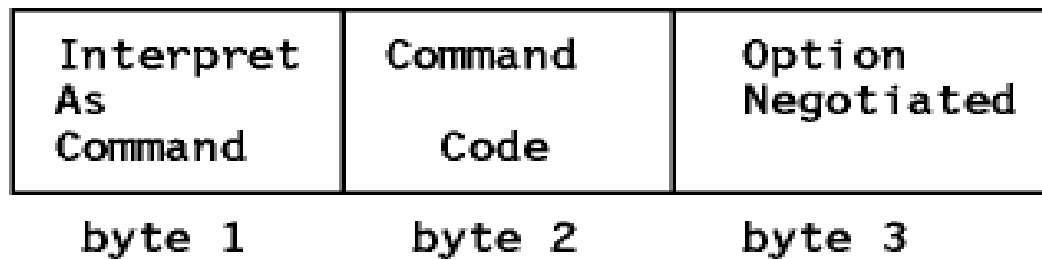
# Network Virtual Terminal

*Via a universal interface called the Network Virtual Terminal (NVT) character set, the TELNET client translates characters (data or commands) that come from the local terminal into NVT form and delivers them to the network. The TELNET server translates data and commands from NVT form into the form acceptable by the remote computer.*

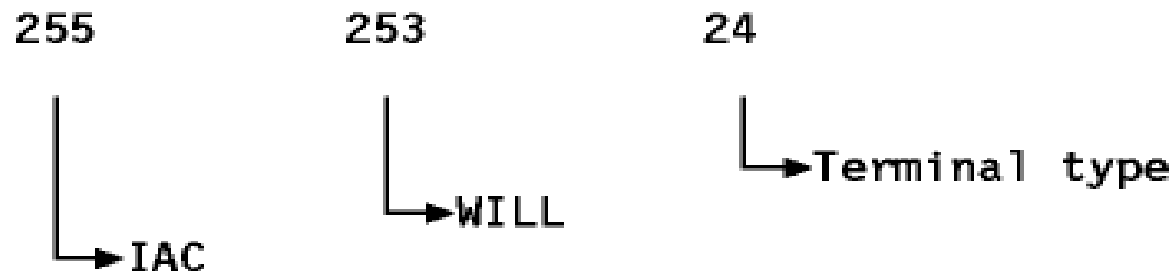
# Network Virtual Terminal

- intermediate representation of a generic terminal.
- provides a standard language for communication of terminal control functions.
- NVT uses two sets of characters, one for data(0) and other for control(1)

# TELNET Command Structure



Sample:



# TELNET Basic Commands

AYT

- Are you There

EC

- Erase Character

AO

- Abort Output

QUIT

- Quit Session

# Outline

Hyper Text Transfer Protocol (HTTP),

FTP,

Email: SMTP, MIME, POP3, Webmail,

Domain Name System (DNS),

TELNET,

**Dynamic Host Control Protocol (DHCP),**

Simple Network Management Protocol (SNMP).

# Need of Dynamic Assignment of IP addresses

## Need of Dynamic assignment of IP addresses

- IP addresses are assigned on-demand
- Avoid manual IP configuration
- Support mobility of laptops
- Support temporary allocation of IP addresses

## Three Protocols:

- RARP (until 1985, no longer used)
- BOOTP (1985-1993)
- DHCP (since 1993)

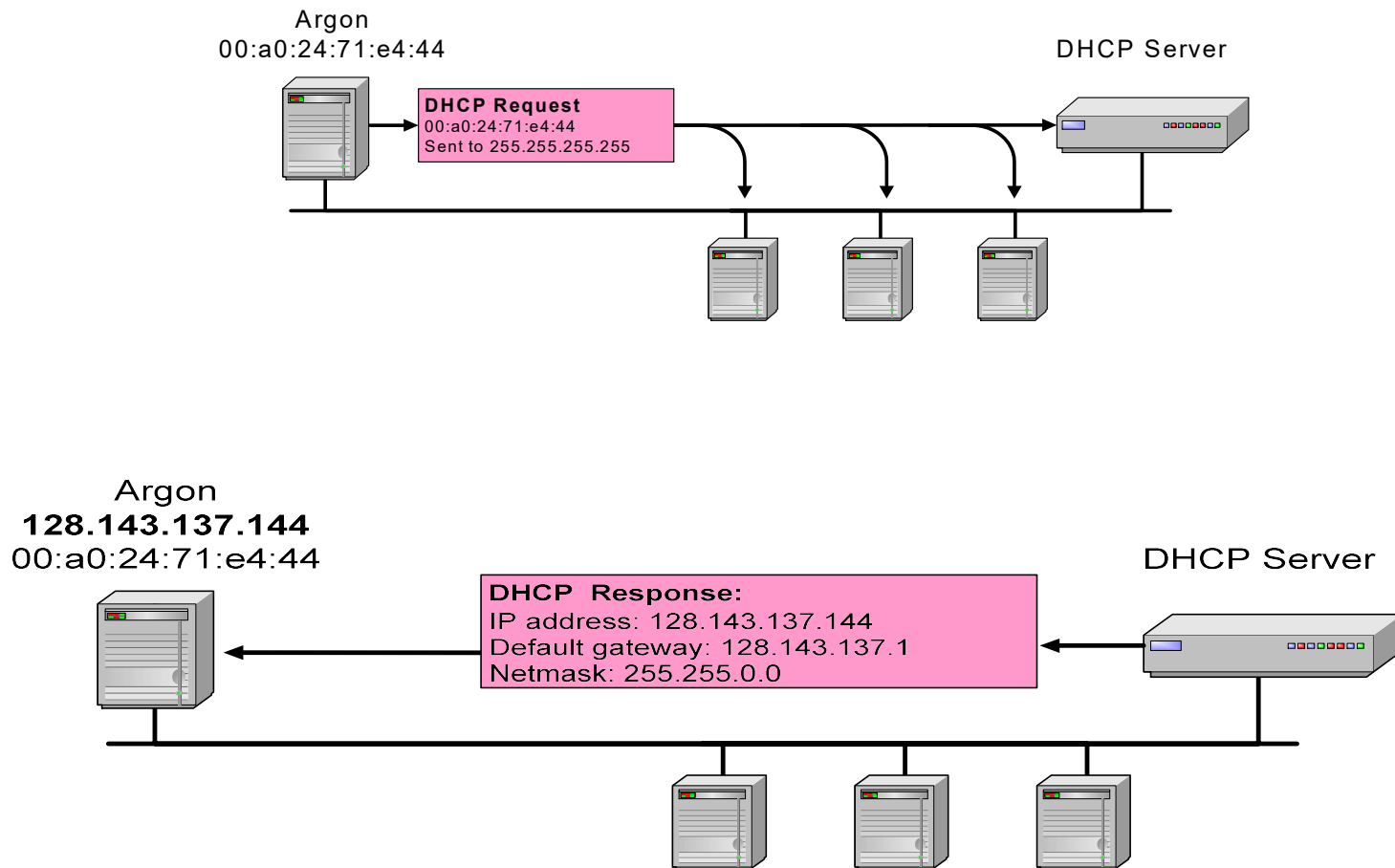
Only DHCP is widely used today.



# DHCP

- Designed in 1993
- Uses UDP
- An extension of BOOTP (Many similarities to BOOTP)
- IP addresses are assigned dynamically on demand
- Supports temporary allocation (“leases”) of IP addresses
- DHCP client can acquire all IP configuration parameters
- DHCP can interoperate with BOOTP clients.

# DHCP Interaction (simplified)



# BOOTP/DHCP Message Format

OpCode	Hardware Type	Hardware Address Length	Hop Count
Number of Seconds		Unused (in BOOTP) Flags (in DHCP)	
Transaction ID			
Client IP address			
Your IP address			
Server IP address			
Gateway IP address			
Client hardware address (16 bytes)			
Server host name (64 bytes)			
Boot file name (128 bytes)			
Options			

(There are >100 different options)

9/19/2017

# BOOTP/DHCP

- **OpCode:** 1 (*Request*), 2(*Reply*)

*Note: DHCP message type is sent in an option*

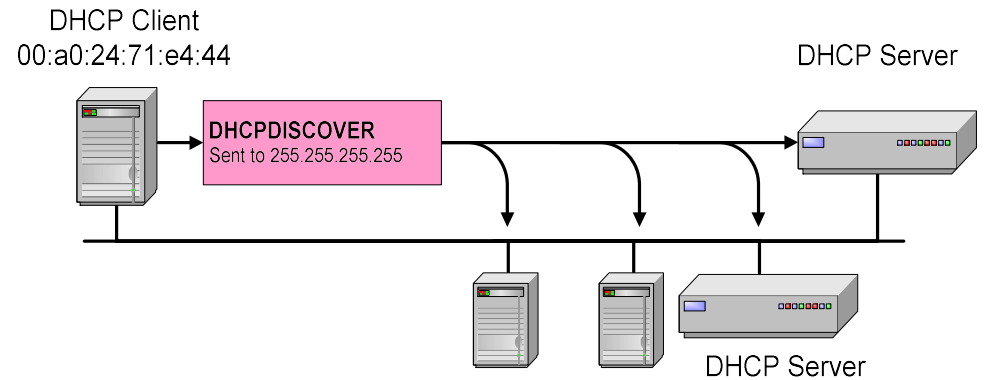
- **Hardware Type:** 1 (*for Ethernet*)
- **Hardware address length:** 6 (*for Ethernet*)
- **Hop count:** *set to 0 by client*
- **Transaction ID:** *Integer (used to match reply to response)*
- **Seconds:** *number of seconds since the client started to boot*
- **Client IP address, Your IP address, server IP address, Gateway IP address, client hardware address, server host name, boot file name:**  
*client fills in the information that it has, leaves rest blank*

# DHCP Message Type

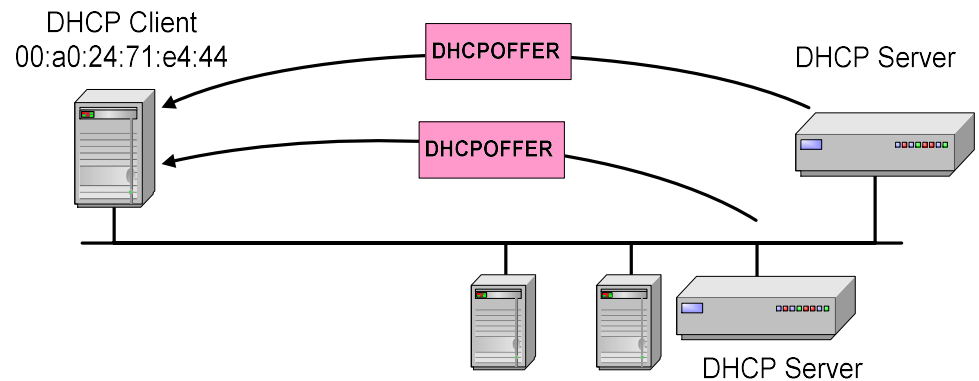
<b>Value</b>	<b>Message Type</b>
1	DHCPDISCOVER
2	DHCPOFFER
3	DHCPREQUEST
4	DHCPDECLINE
5	DHCPACK
6	DHCPNAK
7	DHCPRELEASE
8	DHCPINFORM

# DHCP Operation

- DHCP DISCOVER



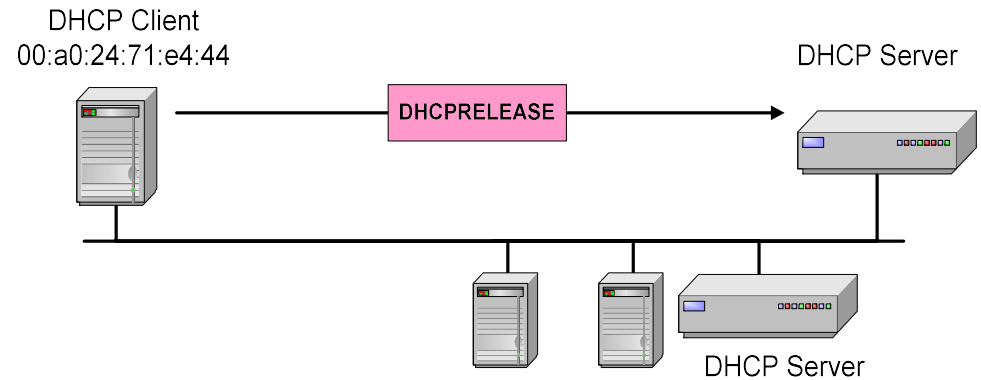
## DCHP OFFER



# DHCP Operation

## DHCP RELEASE

At this time, the DHCP client has released the IP address



# Outline

Hyper Text Transfer Protocol (HTTP),

FTP,

Email: SMTP, MIME, POP3, Webmail,

Domain Name System (DNS),

TELNET,

Dynamic Host Control Protocol (DHCP),

**Simple Network Management Protocol (SNMP).**



# What is Network Management?

Basic tasks that fall under this category are:

- **Configuration Management**

- Keeping track of device settings and how they function

- **Fault Management**

- Dealing with problems and emergencies in the network (router stops routing, server loses power, etc.)

- **Performance Management**

- How smoothly is the network running?
- Can it handle the workload it currently has?

# Functional Areas of Network Management

Configuration Management - inventory, configuration, provisioning

Fault Management - reactive and proactive network fault management

Performance Management - # of packets dropped, timeouts, collisions, CRC errors

Security Management - SNMP doesn't provide much here

Accounting Management - cost management and chargeback assessment

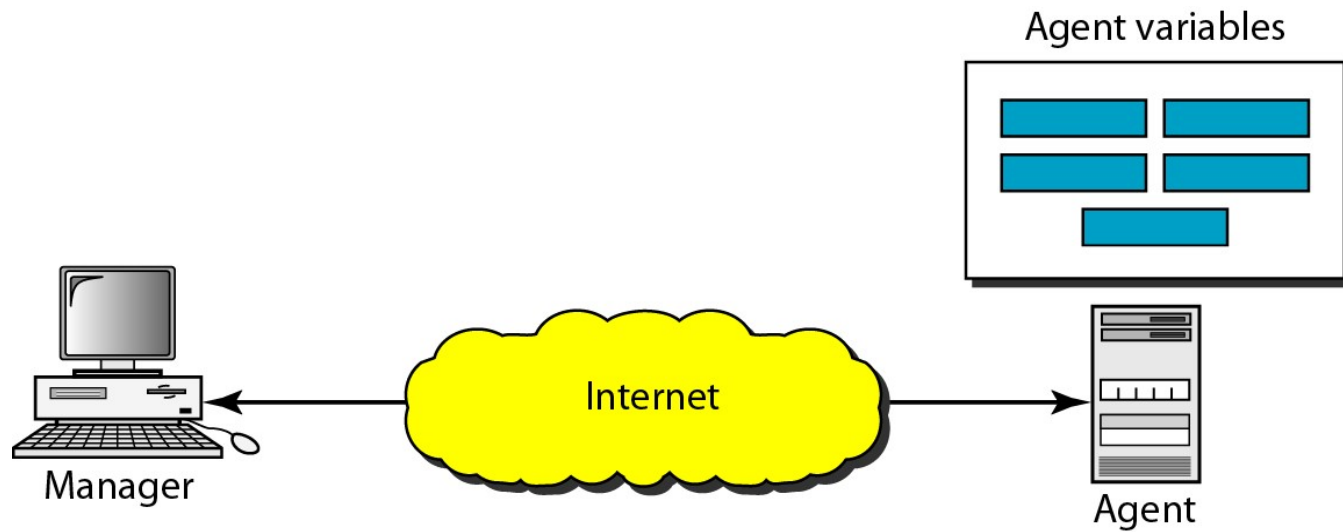
Asset Management - statistics of equipment, facility, and administration personnel

Planning Management - analysis of trends to help justify a network upgrade or  
bandwidth increase

# What is SNMP?

- *SNMP that allows **management of devices on the network** like servers, workstations, routers, switches and other managed devices.*
- Comprised of **agents** and **managers**
  - **Agent** - process running on each managed node collecting information about the device it is running on.
  - **Manager** - process running on a management workstation that requests information about devices on the network.

Figure 28.2 *SNMP concept*



# 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

# Ports & UDP

- SNMP uses User Datagram Protocol (UDP)

- UDP Port 161** - SNMP Messages

- UDP Port 162** - SNMP Trap Messages

CRC

# The Three Parts of SNMP

SNMP network management is based on three parts:

- SNMP Protocol

- Defines format of messages exchanged by management systems and agents.
- Specifies the Get, GetNext, Set, and Trap operations

- Structure of Management Information (SMI)

- Rules specifying the format used to define objects managed on the network that the SNMP protocol accesses

- Management Information Base (MIB)

- List of objects and how they are accessed

---

Figure *Components of network management on the Internet*

---

Management

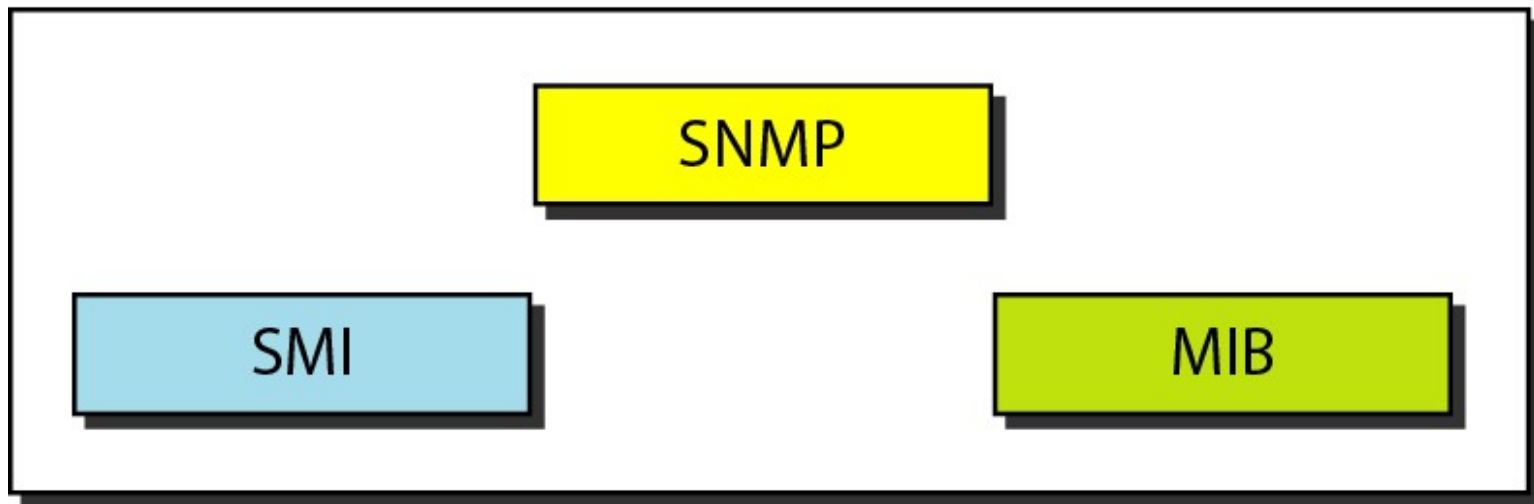
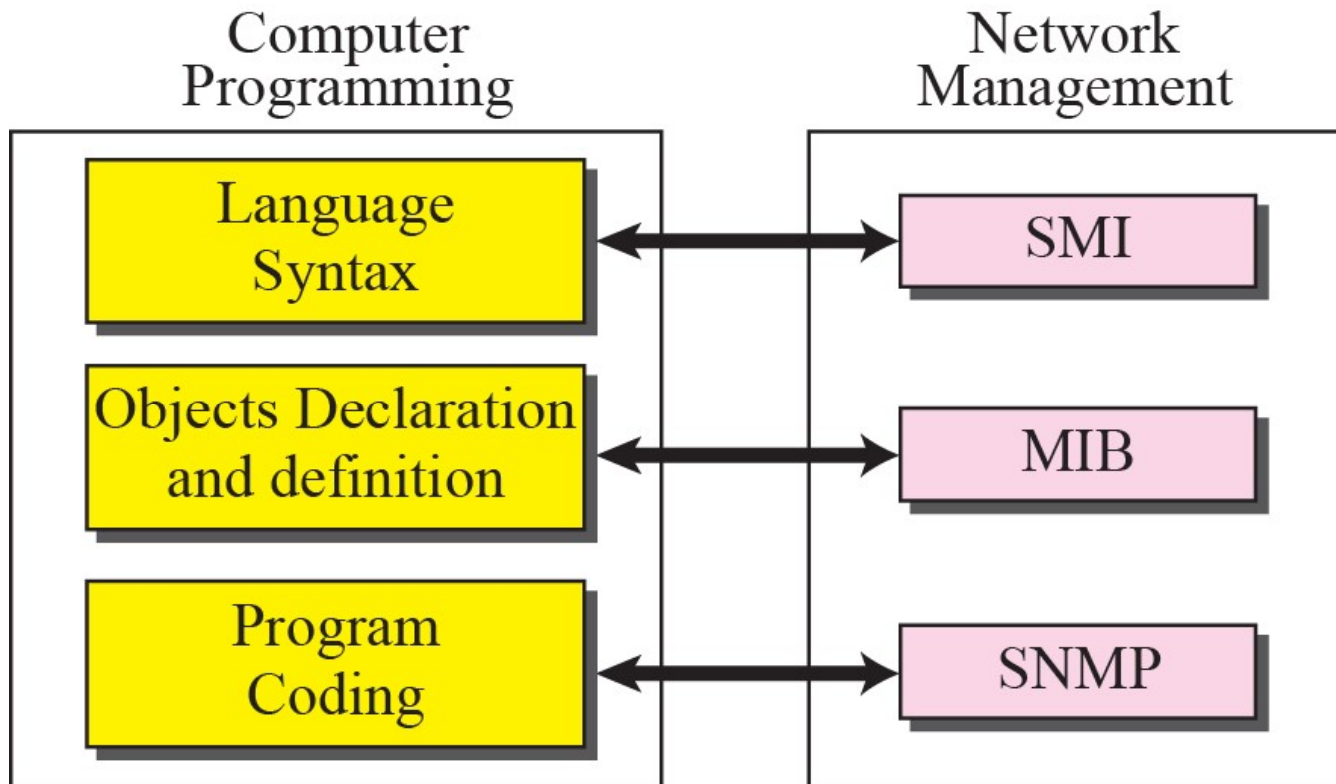




Figure Comparing computer programming and network management



# Four Basic Operations

- **Get**

Retrieves the value of a MIB variable stored on the agent machine  
(integer, string, or address of another MIB variable)

- **GetNext**

Retrieves the next value of the next lexical MIB variable

- **Set**

Changes the value of a MIB variable

- **Trap**

typically a notification of something unexpected, like an error

# Languages of SNMP

- Structure of Management Information (SMI)

specifies the format used for defining managed objects that are accessed via the SNMP protocol

- Abstract Syntax Notation One (ASN.1)

used to define the format of SNMP messages and managed objects (MIB modules) using an unambiguous data description format

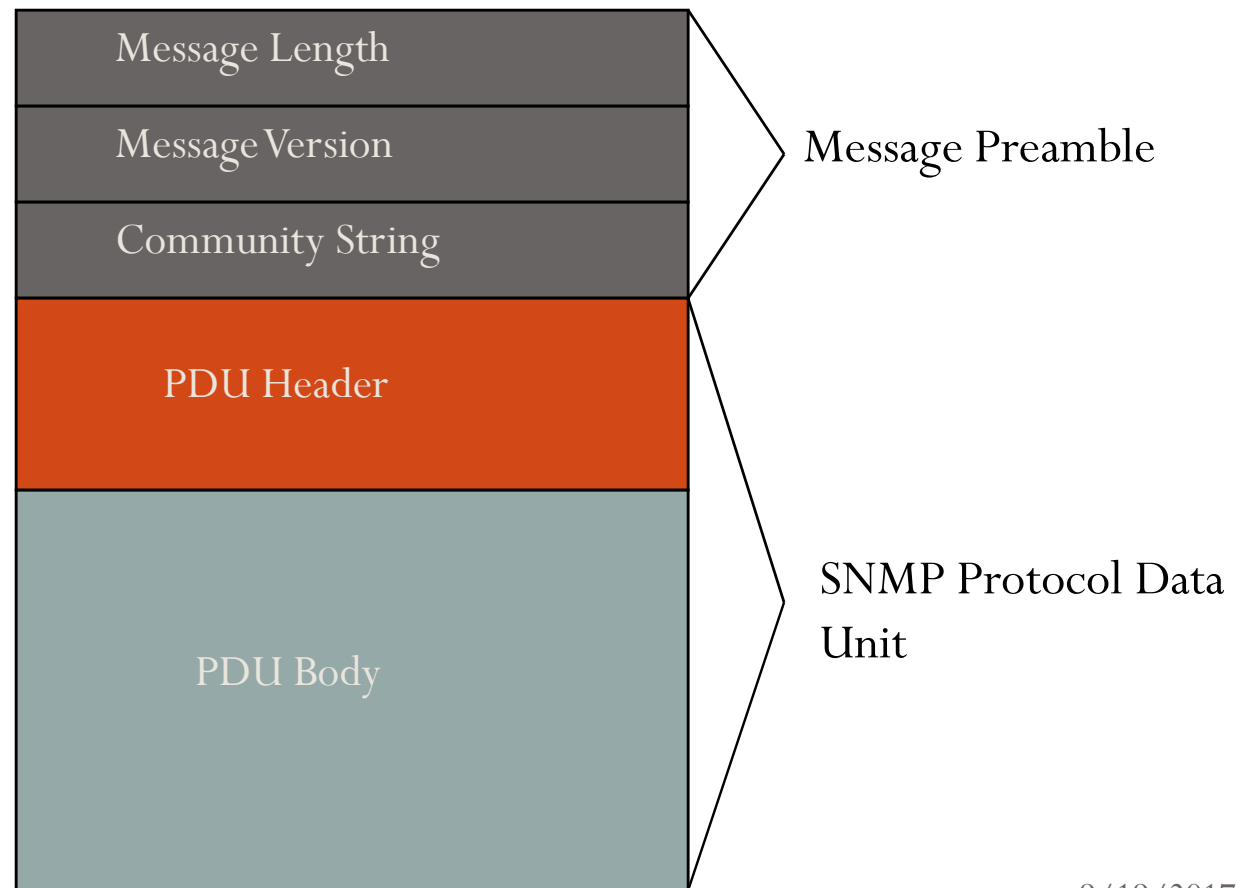
- Basic Encoding Rules (BER)

used to encode the SNMP messages into a format suitable for transmission across a network

# SNMP Data Types

- **INTEGER** -- signed 32-bit integer
- **OCTET STRING**
- **OBJECT IDENTIFIER (OID)**
- **NULL** -- not actually data type, but data value
- **IpAddress** -- OCTET STRING of size 4, in network byte order (B.E.)
- **Counter** -- unsigned 32-bit integer (rolls over)
- **Gauge** -- unsigned 32-bit integer (will top out and stay there)
- **TimeTicks** -- unsigned 32-bit integer (rolls over after 497 days)
- **Opaque** -- used to create new data types not in SNMPv1
- **DateAndTime, DisplayString, MacAddress, PhysAddress, TimeInterval, TimeStamp, TruthValue, VariablePointer** -- textual conventions used as types

# Basic Message Format



# Commercial SNMP Applications

Here are some of the various SNMP Management products available today:

- <http://www.hp.com/go/openview/> HP OpenView
- <http://www.tivoli.com/> IBM NetView
- <http://www.novell.com/products/managewise/> Novell ManageWise
- <http://www.sun.com/solstice/> Sun Microsystems Solstice
- <http://www.microsoft.com/smsmgmt/> Microsoft SMS Server
- <http://www.compaq.com/products/servers/management/> Compaq Insight Manger
- <http://www.redpt.com/> SnmpQL - ODBC Compliant
- <http://www.empiretech.com/> Empire Technologies
- <ftp://ftp.cinco.com/users/cinco/demo/> Cinco Networks NetXray
- <http://www.netinst.com/html/snmp.html> SNMP Collector (Win9X/NT)
- <http://www.netinst.com/html/Observer.html> Observer
- [http://www.gordian.com/products\\_technologies/snmp.html](http://www.gordian.com/products_technologies/snmp.html) Gordian's SNMP Agent
- <http://www.castlerock.com/> Castle Rock Computing
- <http://www.adventnet.com/> Advent Network Management
- <http://www.smplsft.com/> SimpleAgent, SimpleTester

9/19/2017

# References

- [http://higher.ed.mheducation.com/sites/0072460601/student\\_view0/](http://higher.ed.mheducation.com/sites/0072460601/student_view0/)
- [www.wtcs.org/snmp4tpc/files/reference/francois/snmp.ppt](http://www.wtcs.org/snmp4tpc/files/reference/francois/snmp.ppt)
- [ece626web.groups.et.byu.net/Lectures/ch26.ppt](http://ece626web.groups.et.byu.net/Lectures/ch26.ppt)
- <https://www.slideshare.net/MaiyurHossain/http-48129051>