Network Applications

javed shaikh

/agenda for the week

Day 1: Networking Applications

Day 2: Network programming with socket library

Day 3: Introduction to IP Addressing and assistant protocols

Day 4: Transport layer

- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)

Day 5: Network layer

- Internet Protocol (IP)
- Internet Control Message Protocol (ICMP)

/day 1: agenda

- Networking concept
- Networking applications
- Networking application protocols
  - File Transfer Protocol (FTP)
  - Simple Mail Transfer Protocol (SMTP)
  - Hyper Text Transfer Protocol (HTTP)

/networking/concept/1

- *Network* is a set of hardware devices connected together, either physically or logically allowing them to exchange information

- The tasks involved in designing, implementing, upgrading, managing and otherwise working with networks and network technologies, can be termed as *networking*

/networking/concept/2

- Connectivity and communication
- Data sharing
- Hardware sharing
- Internet access
- Internet access sharing
- Data security and management
- Performance enhancement and balancing
- Entertainment

/networking/application/introduction (1)

- Software mechanisms for communication
  - Process to process on same machine
    - Shared memory
    - Pipes
    - Sockets
    - …
  - Process to process on different machines
    - Serial
    - Sockets
    - Transport Library Interface (TLI)

/networking/application/introduction (2)

- Two processes are said to be using computer network when they use networking subsystem (cables, network interface card, switches/hub/routers, network specific software including API) to communicate with each other

/networking/application/introduction (3)

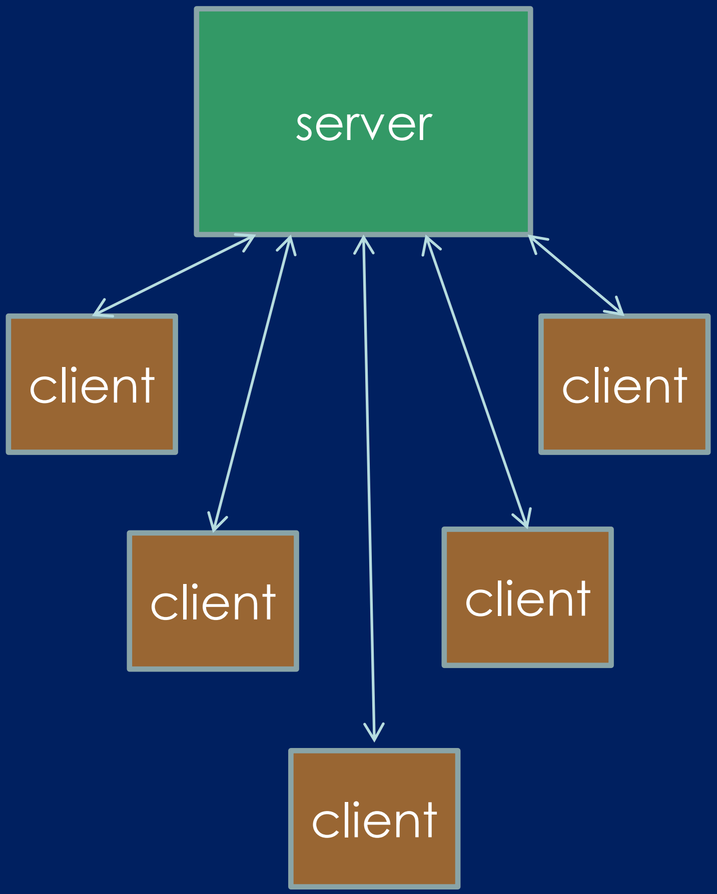
- A network application consists of two parts: client and server
  - Client acts as an active participant
  - Server is considered as a passive communicator
- A situation where both processes have client as well as server capabilities are called peers (equals)

/networking/application/introduction (4)

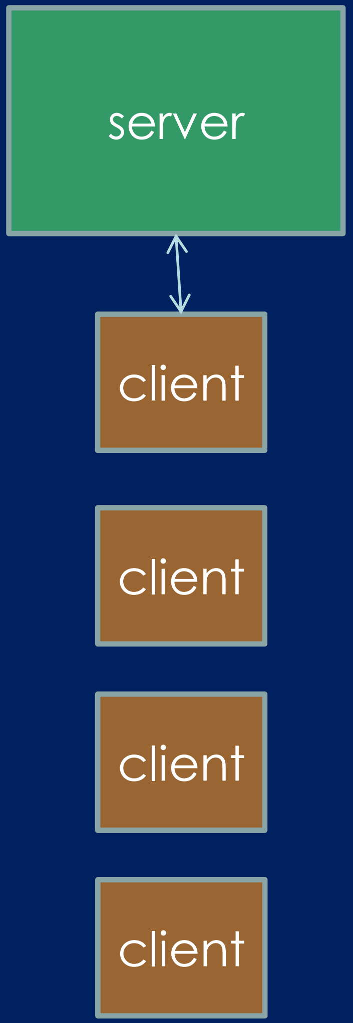
- For two processes to communicate with each other, a mutual understanding is required
- A client-server pair belong to the same application
- In the most simpler network-aware applications typically a client initiates request to the server
- Server responds with appropriate message

client
server

/networking/application/introduction (5)



Concurrent Server

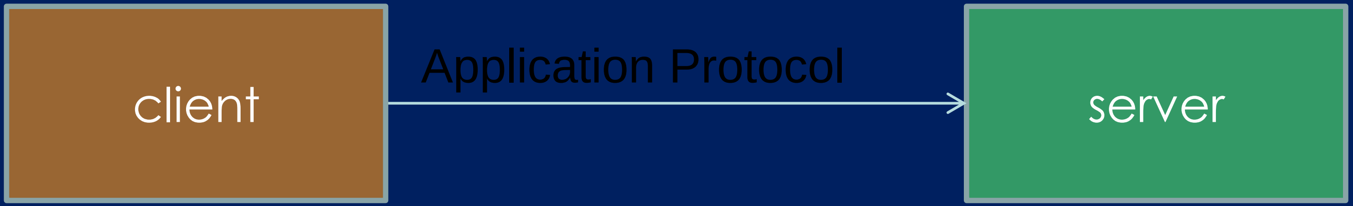


Iterative Server

/networking/application/protocol definition

- The agreement on how client server will communicate with each other is termed as a protocol

- Broadly, a protocol will define requests, responses, terms for negotiations, error messages and behavior under abnormal conditions



/networking/protocol terms

Networking protocols

Internetworking protocols

High-level protocols

Low-level protocols

Protocol stack

Protocol suite

Subprotocols

/networking/application/client-server/E.g. (1)

- Web
  - Purpose: To allow browsing and download of information kept at remote machine
  - Client : Internet browser (e.g. Firefox, IExplorer)
  - Server: A daemon running continuously (e.g. Apache)
  - Protocol: Hyper Text Transfer Protocol (HTTP)
- SSH
  - Purpose: To provide secure login into remote computer
  - Client : ssh Server: sshd
  - Protocol: ?

/networking/application/client-server/E.g. (2)

- FTP
  - Purpose: To allow upload/download of files to/from remote computer
  - Client : ftp program
  - Server: ftpd (A daemon running continuously)
  - Protocol: File Transfer Protocol (FTP)

- Ping
  - Purpose: To check if remote computer is alive
  - Client : ping, Server: part of TCP/IP stack
  - Protocol: Internet Control Message Protocol (ICMP)

/networking/application/peer-peer/E.g.

- talk
  - Purpose: To allow two users to chat
  - Client : talk program, Server : talkd
  - Protocol: ?

- Napster, kazaan
  - Purpose: To exchange multimedia files
  - Client/Server: Same programs
  - Protocol: Proprietary (not available)

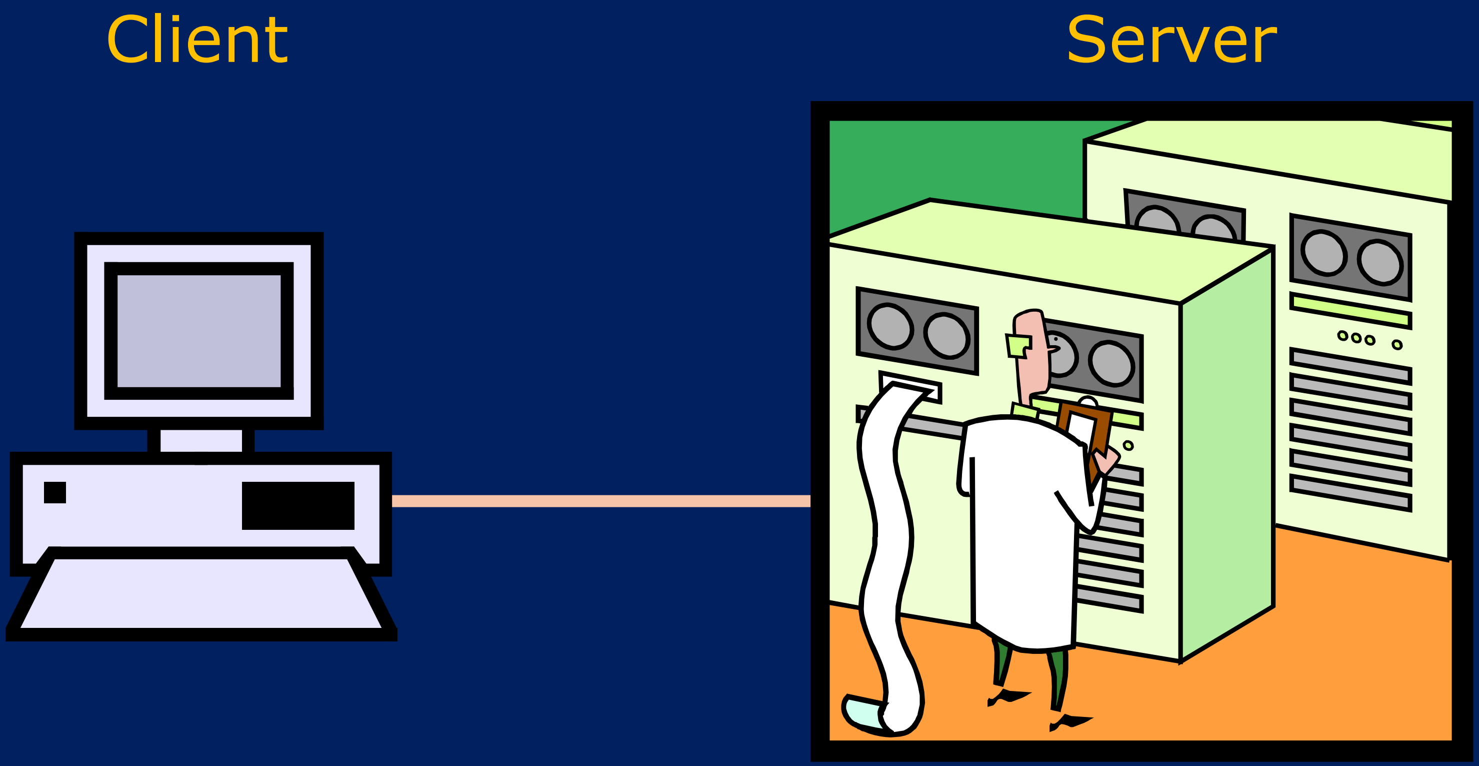
/networking/application/architectures/functions

- 4 possible functions
  - Data storage
  - Data access logic
  - Processing logic
  - Presentation logic

/networking/application/architectures/types

- 3 types
  - Host based
  - Client based
  - Client-Server based

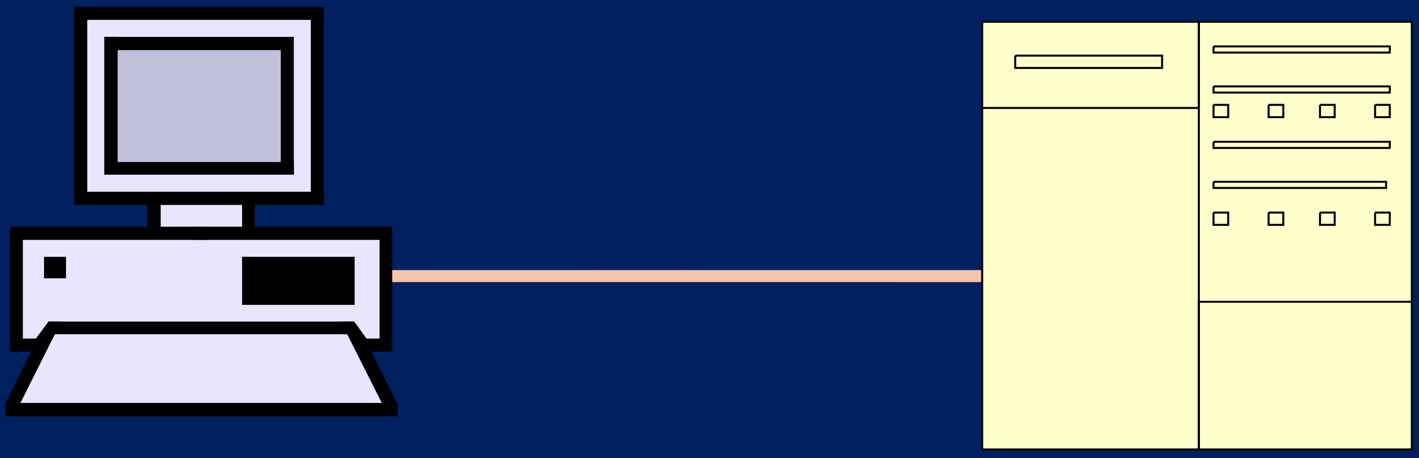
/networking/application/architectures/types



Presentation logic
Application logic
Data access logic
Data storage

Host / Server based

/networking/application/architectures/types



Client

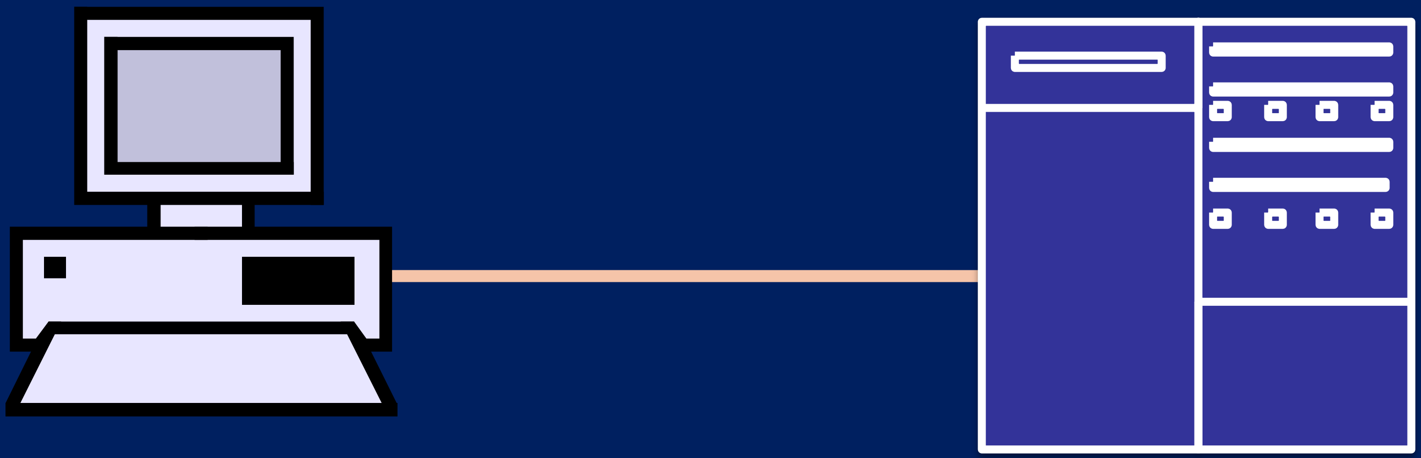
Server

Presentation logic
Application logic
Data access logic

Data storage

Client based

/networking/application/architectures/types



Client

Presentation logic
Application logic

Server

Data access logic
Data storage

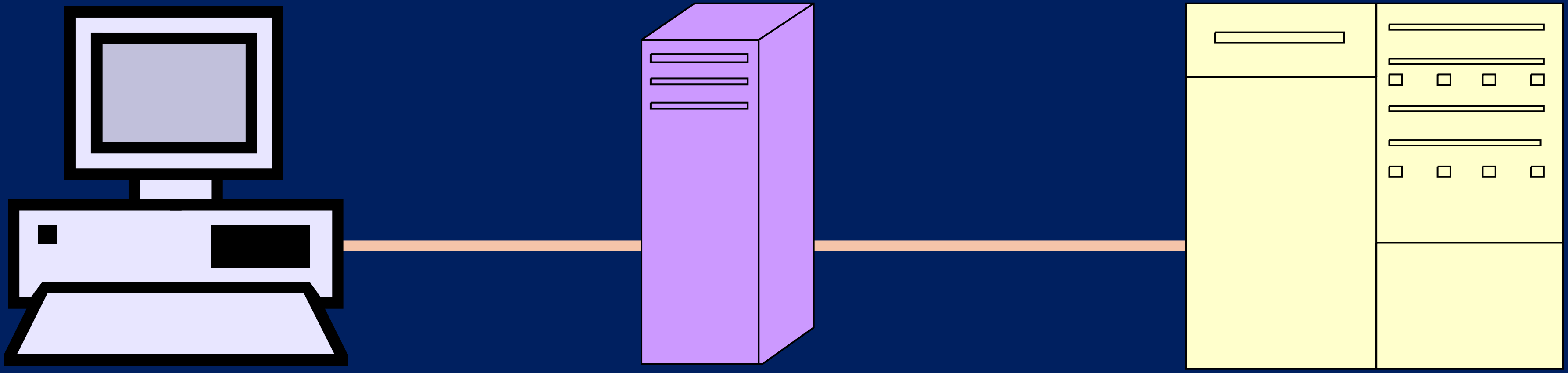
Client-Server

/networking/application/architectures/3-tier

Client

Application Server

Database Server



Presentation logic

Application logic

Data access logic
Data storage

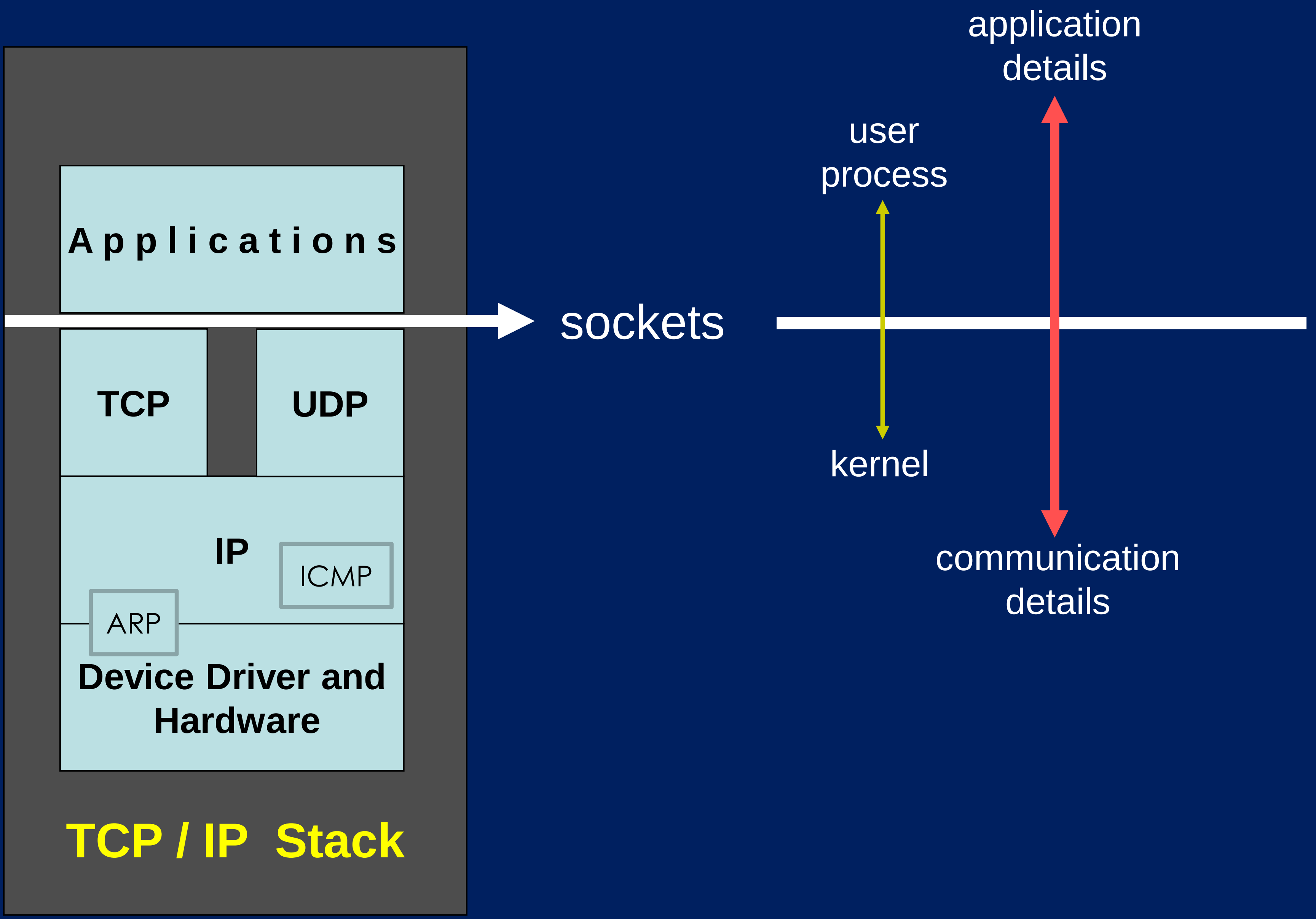
/networking/application/factors

- Cost of Infrastructure – cost of hardware, software, and networks
- Cost of Development – complexity, maintenance, coordination
- Scalability – ability to increase/decrease capacity

/networking/application/factors

	Host-Based	Client-Based	Client-Server
Cost of Infrastructure	High	Medium	Low

/networking/stack

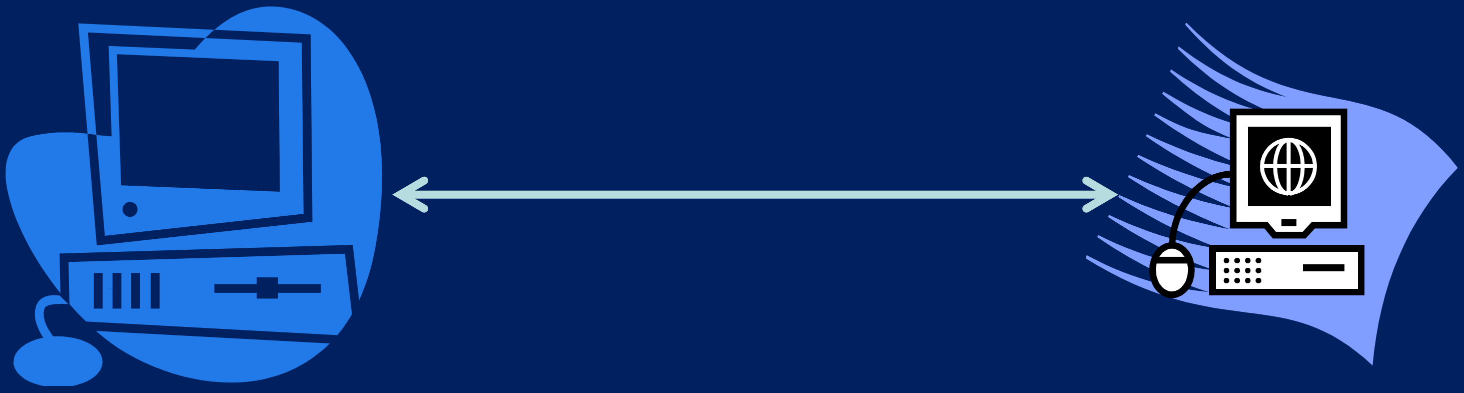


/networking/application/development

- A plan to write network program will comprise of following steps:
  - Requirements
  - Model : client-server or peer-to-peer
  - Protocol : use existing or design?
  - Portability : across different OS
  - Constraints : Internet / Intranet

/networking/application/development/basics

- A socket is a software endpoint that establishes bidirectional communication between a server and client program
- Socket is identified on the Internet by the host's IP address and port number to which it is bound



IP Address
Port number

IP Address
Port number

/networking/application/development/basics

- IP Address : Valid source and destination
- Port number: 1 – 64k
- Transport protocol: TCP / UDP

socket (ip addr, port no)

socket (ip addr, port no)

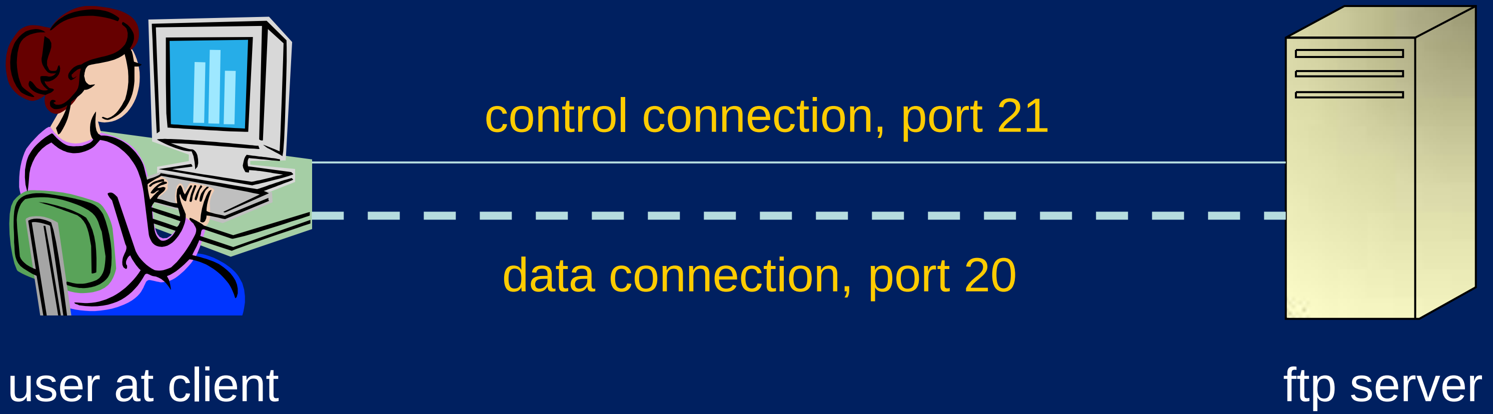
/networking/application/well known protocols

- DNS, DHCP
- Telnet, Rlogin, RSH, SSH
- SMTP
- HTTP
- FTP, TFTP
- NFS
- SNMP
- NNTP, Gopher
- IRC

/networking/application/file transfer

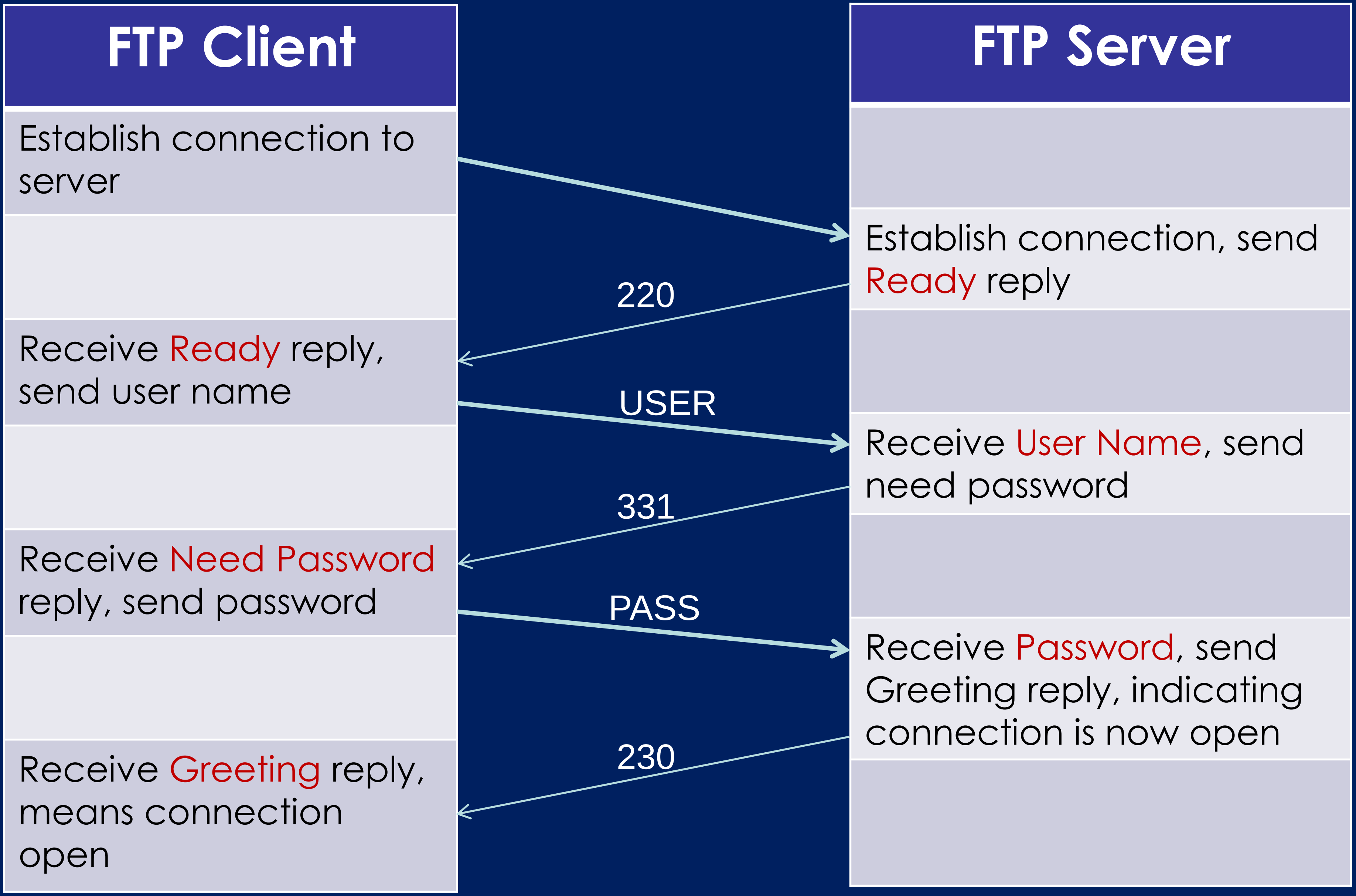
- protocol: File Transfer Protocol (FTP)
- widely used application protocol
- allows efficient transfer of files between two computers
- uses client/server model
- server ports: 21 (control), 20 (data)
- RFC 959 (october 1985)

/networking/application/ftp/details



- Control connection is used throughout the session for passing commands and replies
- A new data connection is opened for every file transfer, and then closed

/networking/application/ftp/details



FTP Client

FTP Server

Establish connection to server

Establish connection, send Ready reply

220

Receive Ready reply, send user name

USER

Receive User Name, send need password

331

Receive Need Password reply, send password

PASS

Receive Password, send Greeting reply, indicating connection is now open

230

Receive Greeting reply, means connection open

/networking/application/ftp/user commands

CMD CODE	Command
USER	User name
PASS	Password
CWD	Change working directory
RETR	Retrieve filename from server
STOR	Send file to server
DELE	Delete filename on server
RMD	Remove dirname on server
MKD	Make dirname on server
PWD	Print working directory
LIST	List all files/dirs in current directory
HELP	Get special help from server (if any)
NOOP	No operation. An ack is expected from server

/networking/application/ftp/reply codes

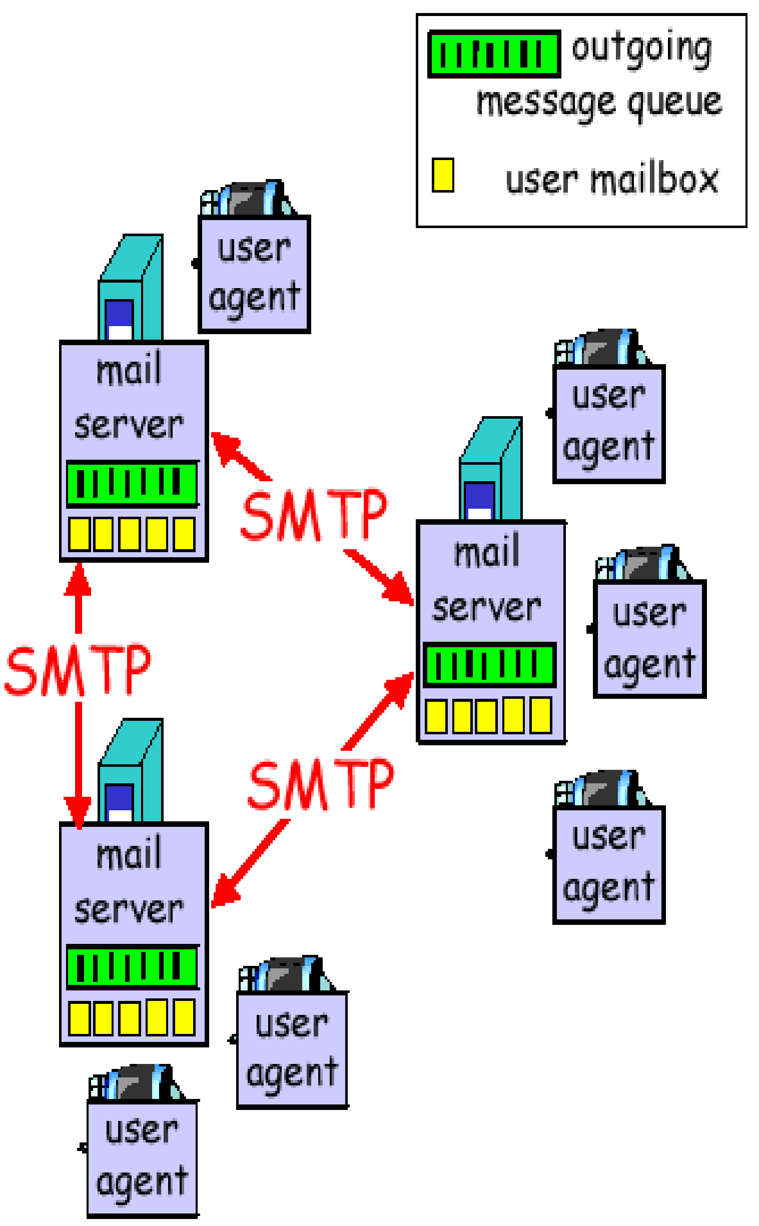
Reply Codes	Meaning
200	Command ok
214	Help message
220	Service ready for new user
221	Service closing control connection
226	Closing data connection
450	Requested file action not taken. File not available
451	Requested action aborted: local error in processing
452	Requested action not taken: Insufficient storage space in system
500	Syntax error, command unrecognized
530	Not logged in

/networking/application/email

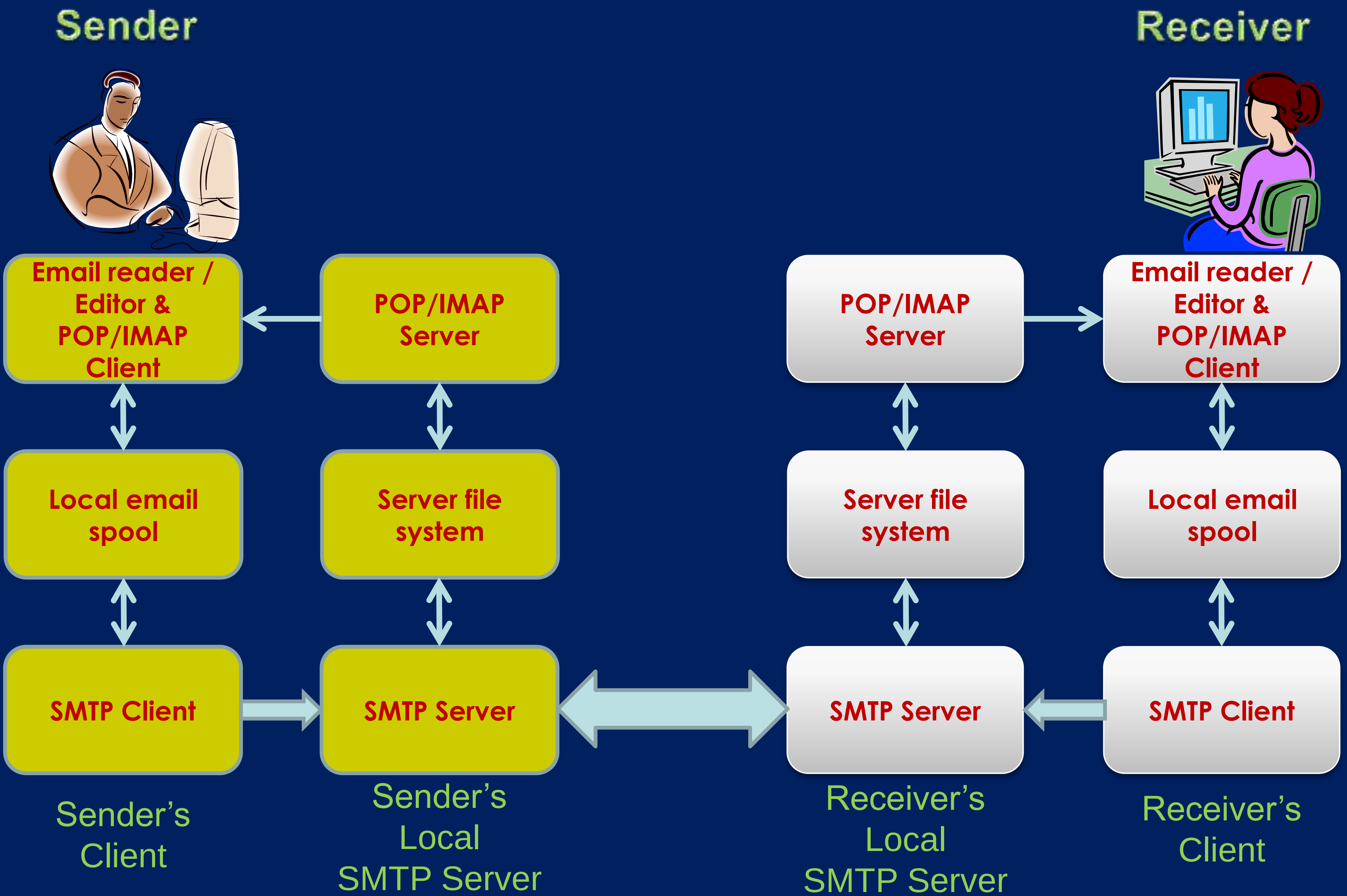
- Protocol: Simple Mail Transfer Protocol (SMTP)
- THE method to communicate today
- Electronic version of Snail mail but instant
- uses client/server model
- server ports: 25 (control), Data ?
- RFC: 2821 (SMTP), 1869 (SMTP Extensions)
- RFC: 2822 (Message standard)

/networking/application/email/steps

- Mail composition
- Mail submission
- Mail delivery
- Mail receipt & processing
- Mail access & retrieval



/networking/application/email/communication



Sender

Receiver

Email reader / Editor & POP/IMAP Client

POP/IMAP Server

Local email spool

Server file system

SMTP Client

SMTP Server

Sender's Client

Sender's Local SMTP Server

POP/IMAP Server

Email reader / Editor & POP/IMAP Client

Server file system

Local email spool

SMTP Server

SMTP Client

Receiver's Local SMTP Server

Receiver's Client

/networking/application/email/communication

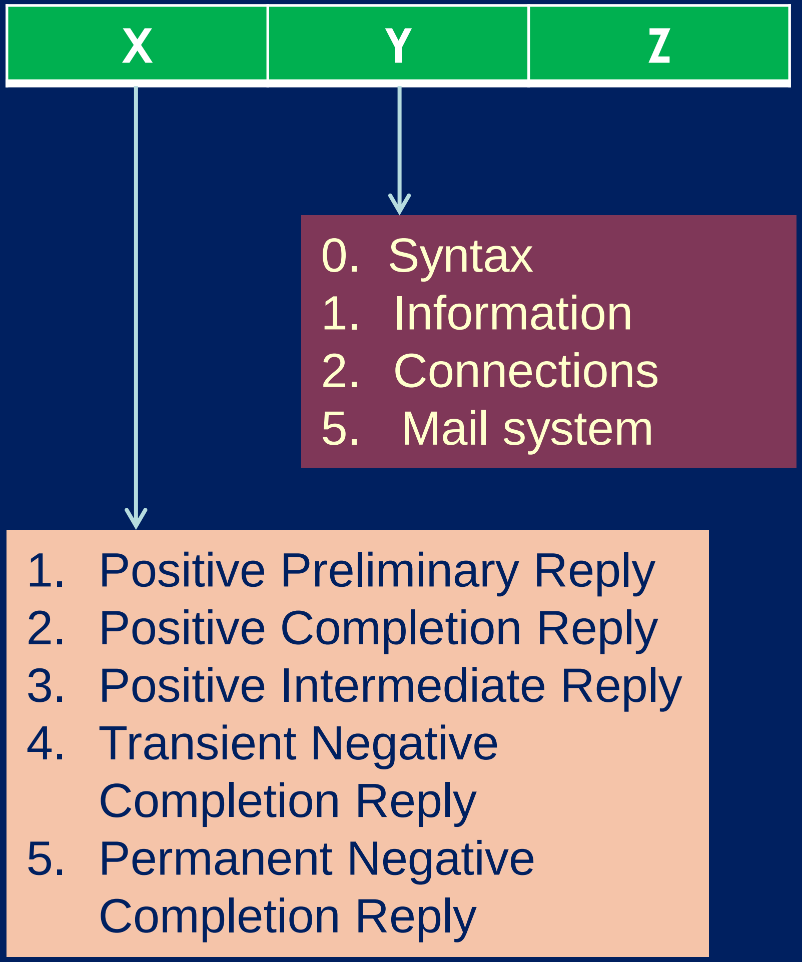
```
S: 220 xyzcompany.com
C: HELO acts.cdac.in
S: 250 Hello acts.cdac.in, pleased to meet you
C: MAIL FROM: <me@acts.cdac.in>
S: 250 me@acts.cdac.in ... Sender ok
C: RCPT TO: <hr@xyzcompany.com>
S: 250 hr@xyzcompany.com ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Dear HR Manager,
C: I would like to ….
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 xyzcompany.com closing connection
```

/networking/application/email/SMTP commands

Command Code	Command	Parameters
HELO	Hello	Sender’s domain name
EHLO	Extended Hello	Extension list is exchanged
MAIL	Initiate mail transaction	FROM
RCPT	Recipient	TO
DATA	Mail message data	
RSET	Reset the connection	
VRFY	Verify	Email addr to be verified
EXPN	Expand	Email addr of mailing list
HELP	Help	(Optional)
NOOP	No Operation	
QUIT	Quit	(Terminates the SMTP session)

/networking/application/email/SMTP reply codes

Reply Code	Reply Text
220	<servername> Service ready
221	<servername> closing transmission channel
354	Start mail input. End with <CRLF>.<CRLF>
421	<servername> Service not available, closing transmission channel
452	Requested action not taken: Insufficient system storage
500	Syntax error, command unrecognized
552	Requested mail action aborted: exceeded storage allocation



0. Syntax
1. Information
2. Connections
5. Mail system

1. Positive Preliminary Reply
2. Positive Completion Reply
3. Positive Intermediate Reply
4. Transient Negative Completion Reply
5. Permanent Negative Completion Reply

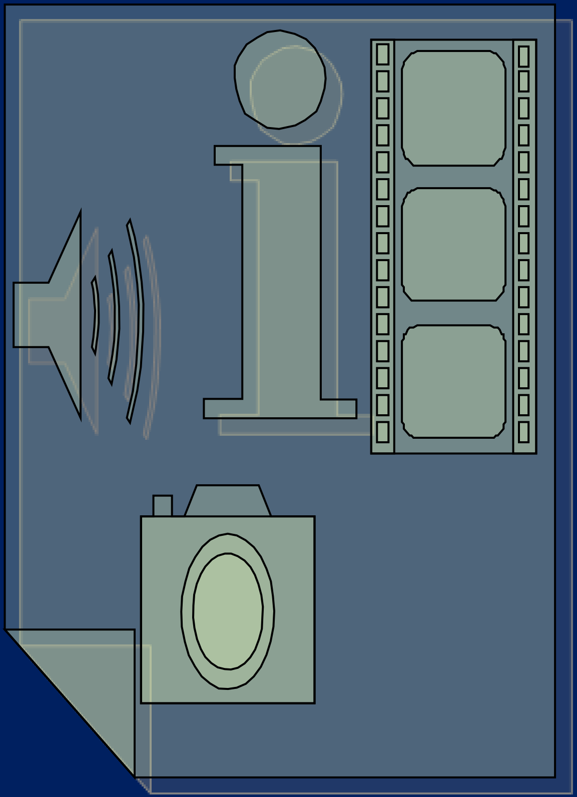
/networking/application/email/header fields

Field Name	Requirement
Date:	Mandatory
From:	Mandatory
Sender:	Optional
Reply-To:	Optional
To:	Normally present
Cc:	Optional
Bcc:	Optional
Message-ID:	Should be present
In-Reply-To:	Optional
References:	Optional
Subject:	Normally present
Comments:	Optional
Keywords:	Optional
Received: Return-Path:	Inserted by email system

/networking/application/world wide web

- protocol: Hyper Text Transfer Protocol
- single most important application in the history of networking
- allows user to move around documents and related information from one location to other
- uses client/server model
- server port: 80
- RFC 2616 (HTTP 1.1, 1999)

/networking/application/world wide web

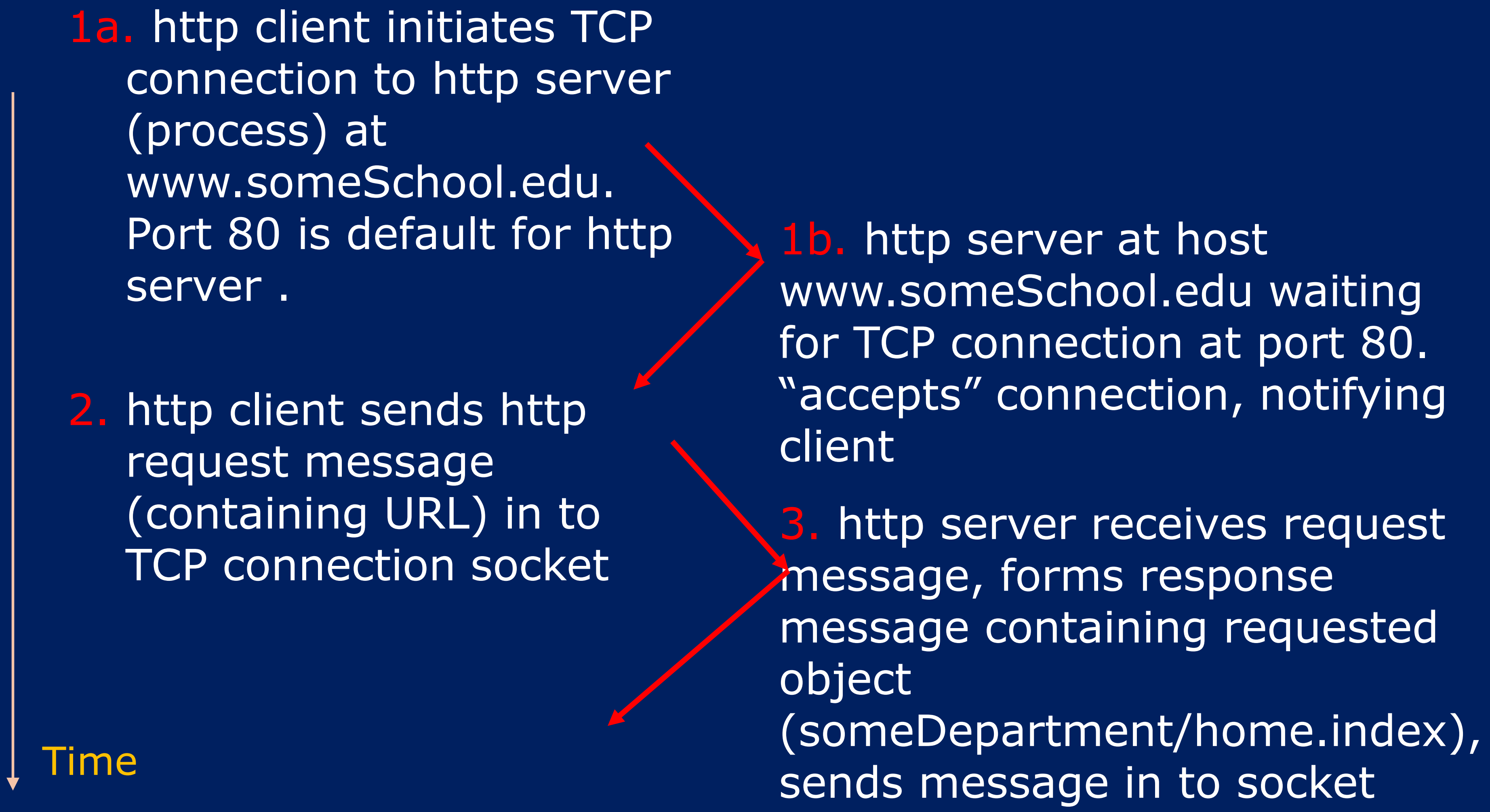


- Webpage:
  - consists of “ objects”
  - addressed by a URL
- Most Web pages consist of :
  - base HTML page , and
  - several referenced objects
- URL has two components
  - host name and path name
- User agent for Web is called a browser:
  - MS Internet Explorer , Firefox, Opera
- Server for Web is called Web server :
  - Apache (public domain)
  - MS Internet Information Server

/networking/application/world wide web

http://www.someSchool.edu/someDepartment/home.index

(Contains text and referance to 10 jpeg images)



1a. http client initiates TCP connection to http server (process) at www.someSchool.edu. Port 80 is default for http server .

1b. http server at host www.someSchool.edu waiting for TCP connection at port 80. “accepts” connection, notifying client

2. http client sends http request message (containing URL) in to TCP connection socket

3. http server receives request message, forms response message containing requested object (someDepartment/home.index), sends message in to socket

/networking/application/world wide web

http://www.someSchool.edu/someDepartment/home.index

(Contains text and reference to 10 jpeg images)

4. http server closes TCP connection .

5. http client receives response message containing html file, displays html. On parsing html file, finds 10 referenced jpeg objects.

6. Steps 1 – 5 repeated for each of 10 jpeg objects

Time

http is “stateless”
server maintains no information about past client requests

Protocols that maintain “state” are complex !

Transitory connections	Persistent connections
- HTTP/1.0	- Default for HTTP/1.1
- Server parses request, responds and closes TCP connection	- On same TCP connection : server parses request, responds, parses new request, …
- 2RTTs to fetch each object	- Client sends requests for all referenced objects as soon as it receives base HTML
- Each object transfer suffers from slow start	- Fewer RTTs and less slow start

/networking/application/world wide web

Status codes	Explanation
202	OK. Request succeeded, requested object later in this message
301	Moved Permanently Requested object moved, new location specified later in this message (Location:)
400	Bad Request Requested message not understood by server
404	Not Found Requested document not found on this server
505	HTTP Version Not Supported

Questions?