



Indian Institute of Technology Kharagpur

Client Server Concepts, DNS, Telnet, FTP



Lecture 9: Client server concepts, DNS, Telnet, FTP

On completion, the student will be able to:

- **Explain how the client-server concept works.**
- **Explain and illustrate the name resolution process used by DNS servers.**
- **Explain how the Telnet and FTP protocol works.**
- **Demonstrate usage of Telnet and FTP commands.**

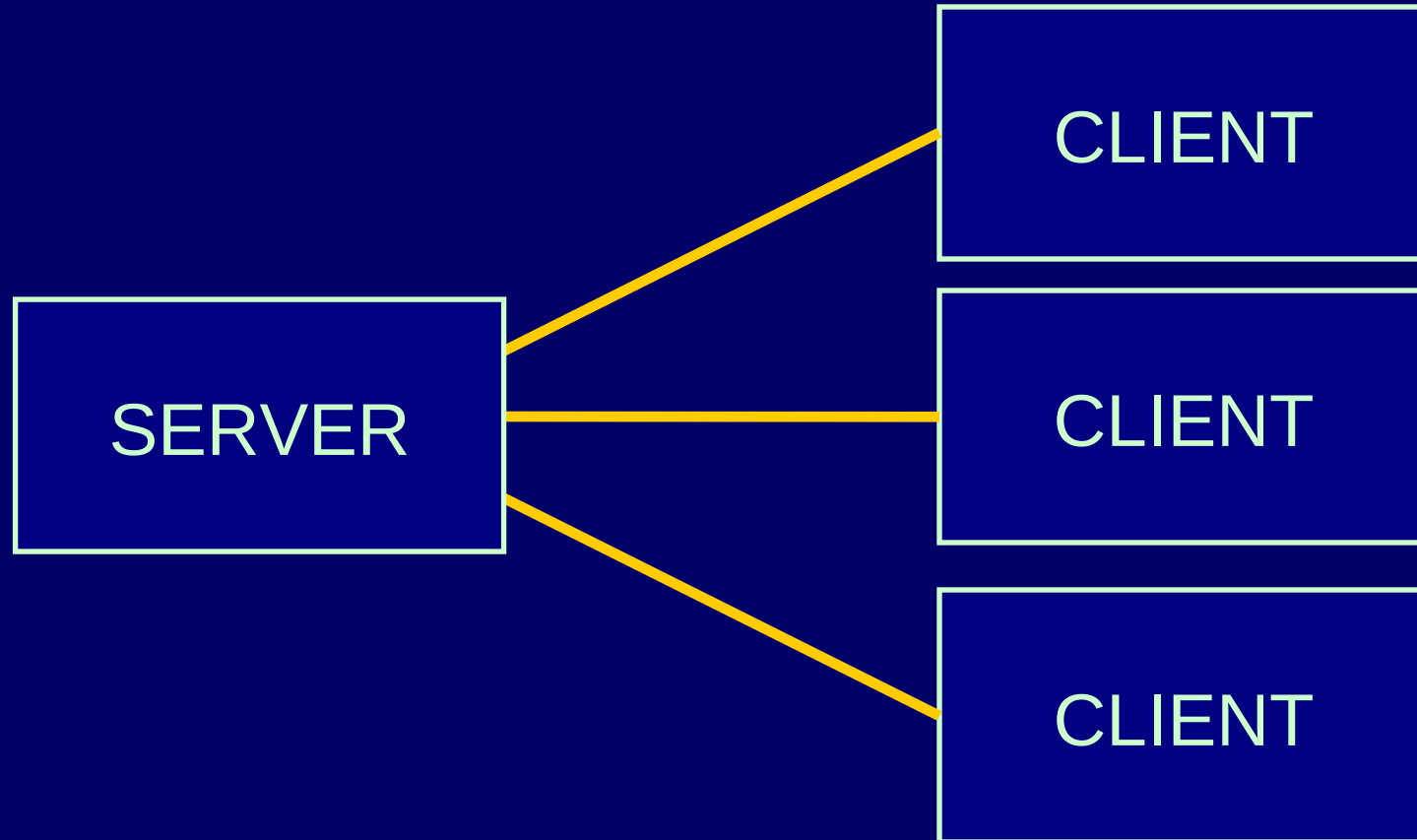


Client-server Model

- **Standard model for developing network applications.**
- **Notion of client and server.**
 - **A server is a process that is offering some service.**
 - **A client is a process that is requesting the service.**
 - **Server or client may be running on different machines.**
 - **Server waits for requests from client(s).**



Client-server Model (contd.)





Client-server Model (contd.)

- **Typical scenario:**
 - **The server process starts on some computer system.**
 - **Initializes itself, then goes to sleep waiting for a client request.**
 - **A client process starts, either on the same system or on some other system.**
 - **Sends a request to the server.**



- When the server process has finished providing its service to the client, the server goes back to sleep, waiting for the next client request to arrive.
- The process repeats.



Client-server Model (contd.)

- Roles of the client and the server processes are asymmetric.
- Two types of servers:
 - *Iterative servers.*
 - *Concurrent servers.*



Iterative Servers

- Used when the server process knows in advance how long it takes to handle each request and it handles each request itself.
 - Single copy of server runs at all times.
 - A client may have to wait if the server is busy.



Concurrent Servers

- Used when the amount of work required to handle a request is unknown; the server starts another process to handle each request.
 - A copy of the server caters to a client's request in a dedicated fashion.
 - As many copies of server as there are client requests.



Using TCP or UDP

- Before start of communication, a connection has to be established between the two hosts.
- Five components in a connection:
 - Protocol used
 - Source IP address
 - Source port number
 - Destination IP address
 - Destination port number



Develop a Network Application

- The best way is to use some standard and well-accepted protocol.
 - At the data link layer level, use **Ethernet**.
 - At the network layer level, use **IP**.
 - At the transport layer level, use **TCP**.
 - At the application layer level, use a standard API like the **Berkeley Socket Interface**.



What is a Socket?

- The *socket* is the BSD method for achieving inter-process communication (IPC).
- It is used to allow one process to speak to another (on same or different machine).
 - *Analogy*: Like the telephone is used to allow one person to speak to another.



Basic Idea

- When two processes located on two machines communicate, we define association and socket.
 - **Association:** basically a 5-tuple
 - Protocol
 - Local IP address
 - Local port number
 - Remote IP address
 - Remote port number



- **Socket: also called half-association (a 3-tuple)**
 - **Protocol, local IP address, local port number**
 - **Protocol, remote IP address, remote port number**



Domain Name System (DNS)



What is DNS?

- The global database system for Internet addressing, mail and other information.
 - Much easier to use and memorize.
- Concept of domains and sub-domains.
 - Domain management is distributed.
 - DNS servers translate domain names to IP addresses.



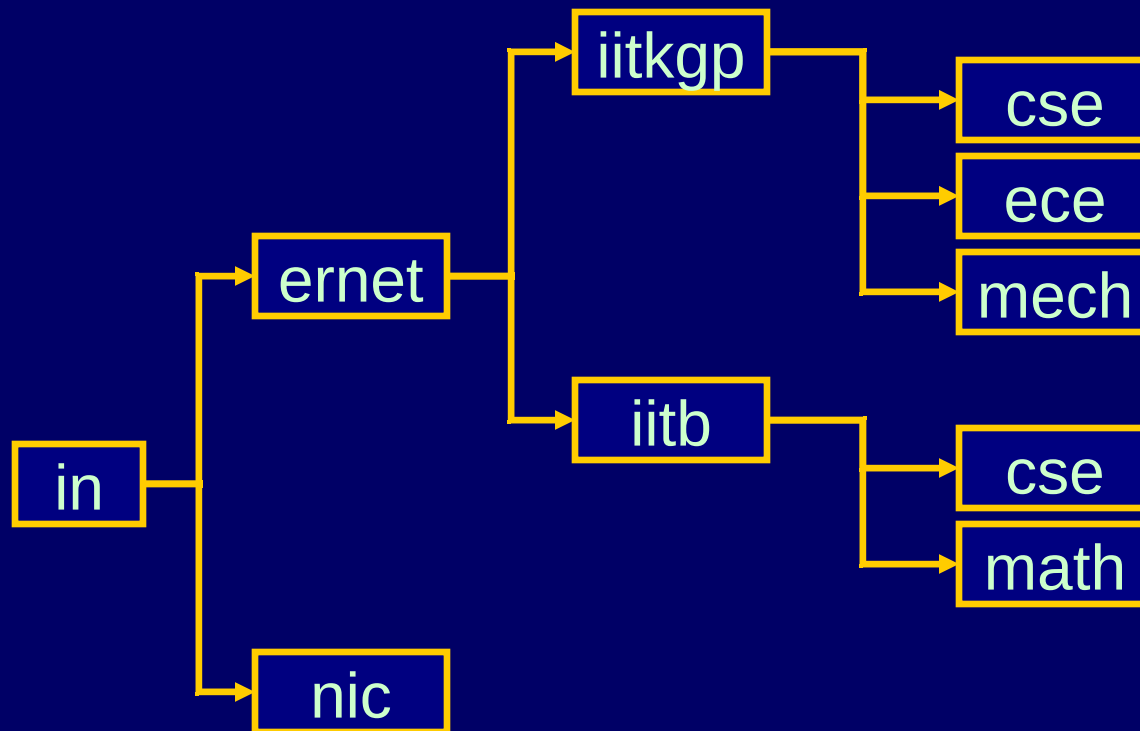
Top Level Domains

- **com** – Commercial
- **org** – Non-profit
- **net** – Network service provider
- **gov** – US govt.
- **mil** – military
- **edu** – Education
- **au** – Australian
- **at** – Austrian
- **ca** – Canadian
- **dk** – Dutch
- **fr** – French
- **de** – German
- **in** – Indian
- **it** – Italian
- **jp** – Japanese
- **kr** – Korean
- **nz** – New Zealand
- **es** – Spanish
- **tw** – Taiwanese
- **uk** – British or Irish
- **us** – U.S.



Domain Name Structure

- Domain names are arranged in a hierarchical tree-like structure.



Eg: cse.iitkgp.ernet.in



Some Statistics

In thousands

Year	COM	EDU	GOV	MIL	NET	ORG
1991	200	250	75	20	5	50
1992	350	450	100	20	10	50
1993	700	700	150	30	25	75
1994	1,200	1,075	250	100	125	175
1995	2,400	1,700	300	250	700	250



Name Resolution Process

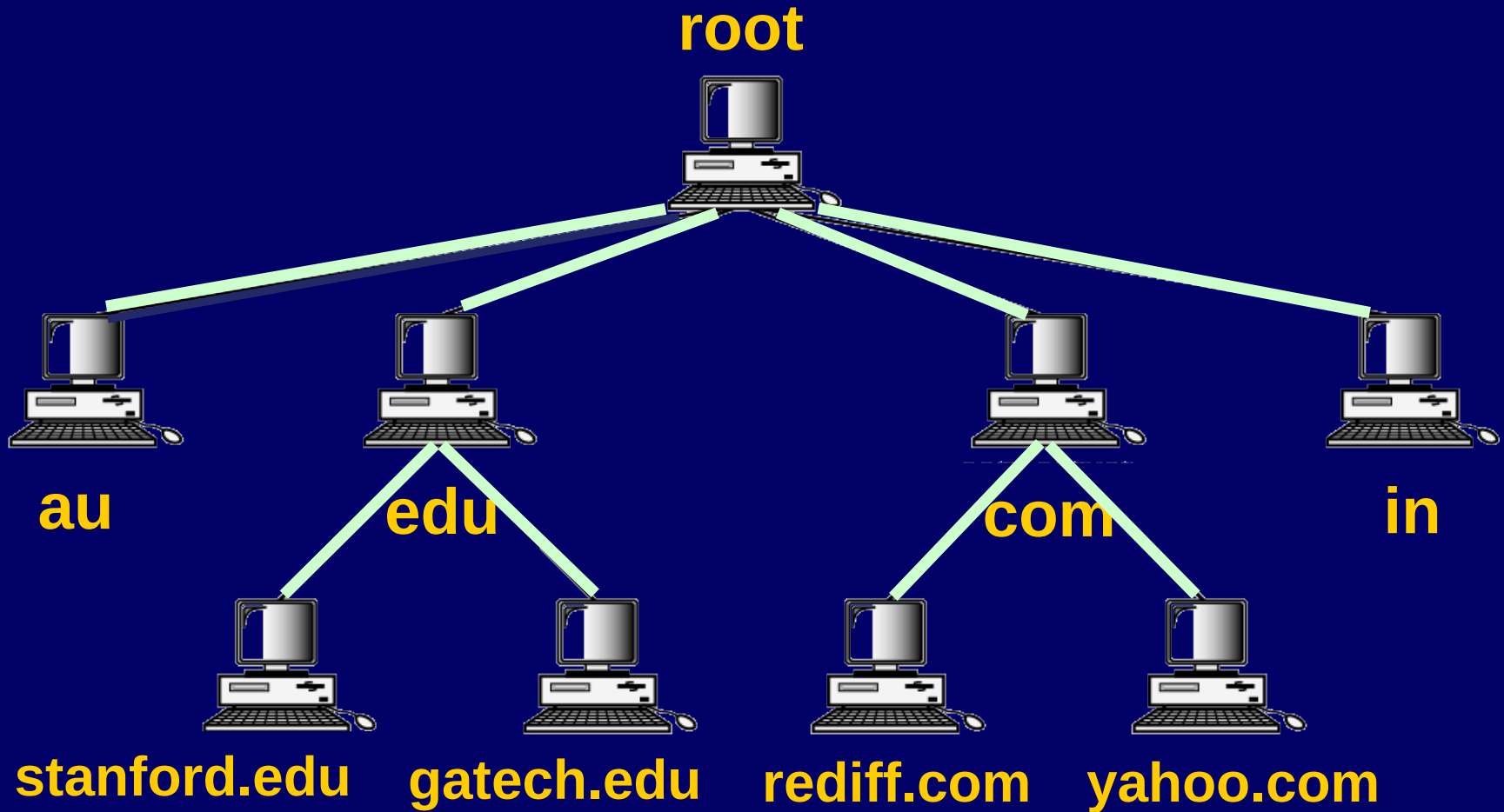
- The commonly used server is BIND (Berkeley Internal Name Domain).
 - Runs under UNIX as a process called **named**.
- When an application needs some information from the server, it invokes the DNS name resolver.
 - DNS translates a fully qualified domain name into the corresponding IP address.
 - Using the command **nslookup**.



- If the name server does not have the information locally, it asks its primary server, and so on.
- For redundancy, each host may also have one or more secondary name servers which may be queried when the primary fails.
- How do name servers update themselves?

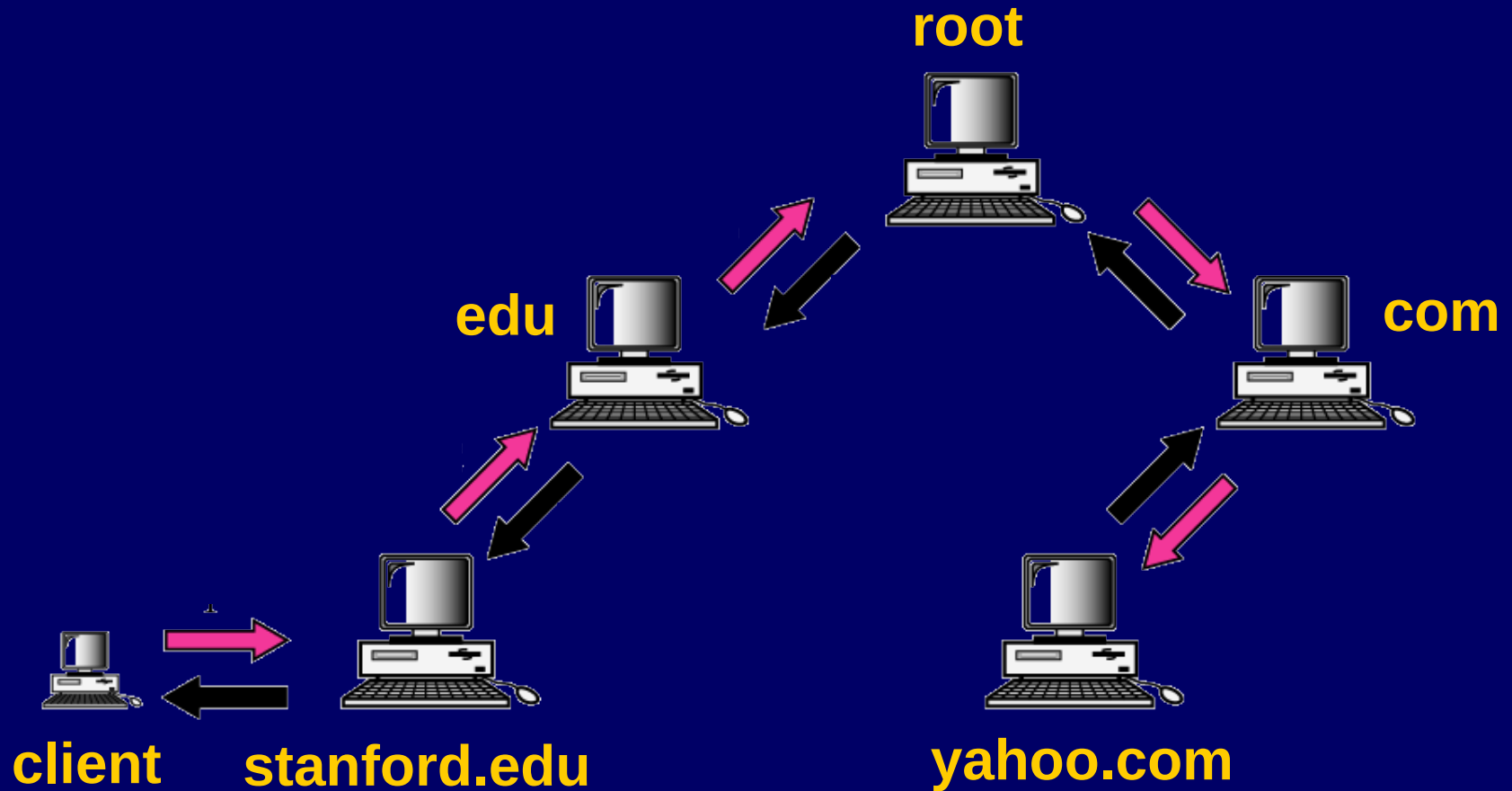


Hierarchy of Name Servers





Recursive Name Resolution





Iterative Name Resolution

- Client sequentially sends queries to DNS servers and receives response.
 - If response is negative, the DNS server to query next is also returned.
 - Unlike recursive name resolution, where only one response is finally returned back to the client.

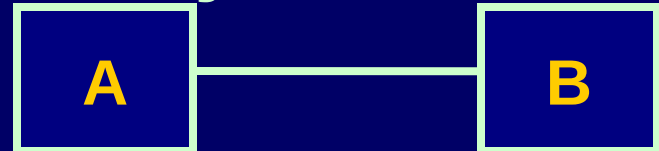


TELNET Protocol



TELNET

- Allows a person sitting on one computer to work on another computer.
- Starts a remote session on another machine.
- Requires a person to supply login name and password to gain entry.
- Command:
 - telnet <domain_name>
 - telnet <ip_address>





TELNET (contd.)

- **Why required?**
 - For using software only available on the remote host.
 - For using devices (like printer) connected to the remote host.
- **Typical scenario**
 - Many users do a telnet to a remote server, and work there.
 - Server is a bigger and faster computer.



TELNET (contd.)

- By default connection is established over port 23.
- Any other port number can also be specified.

```
telnet cse.iitkgp.ernet.in 25
```



A Sample TELNET Session

```
SUNNY>> telnet cse
Trying 144.16.192.57...
Connected to cse.
Escape character is '^]'.
Linux 1.2.8 (cse) (ttyp0)
```

```
cse login: isg
Password:
Last login: Wed Jun 11 12:17:55 from
144.16.202.210
Linux 1.2.8. (POSIX).
You have new mail.
cse:~$
```



FTP Protocol



File Transfer Protocol (FTP)

- Allows a person to transfer files between two machines.
- Requires a person to supply login name and password to gain entry.
- Command:

`ftp <domain_name>`

`ftp <ip_address>`

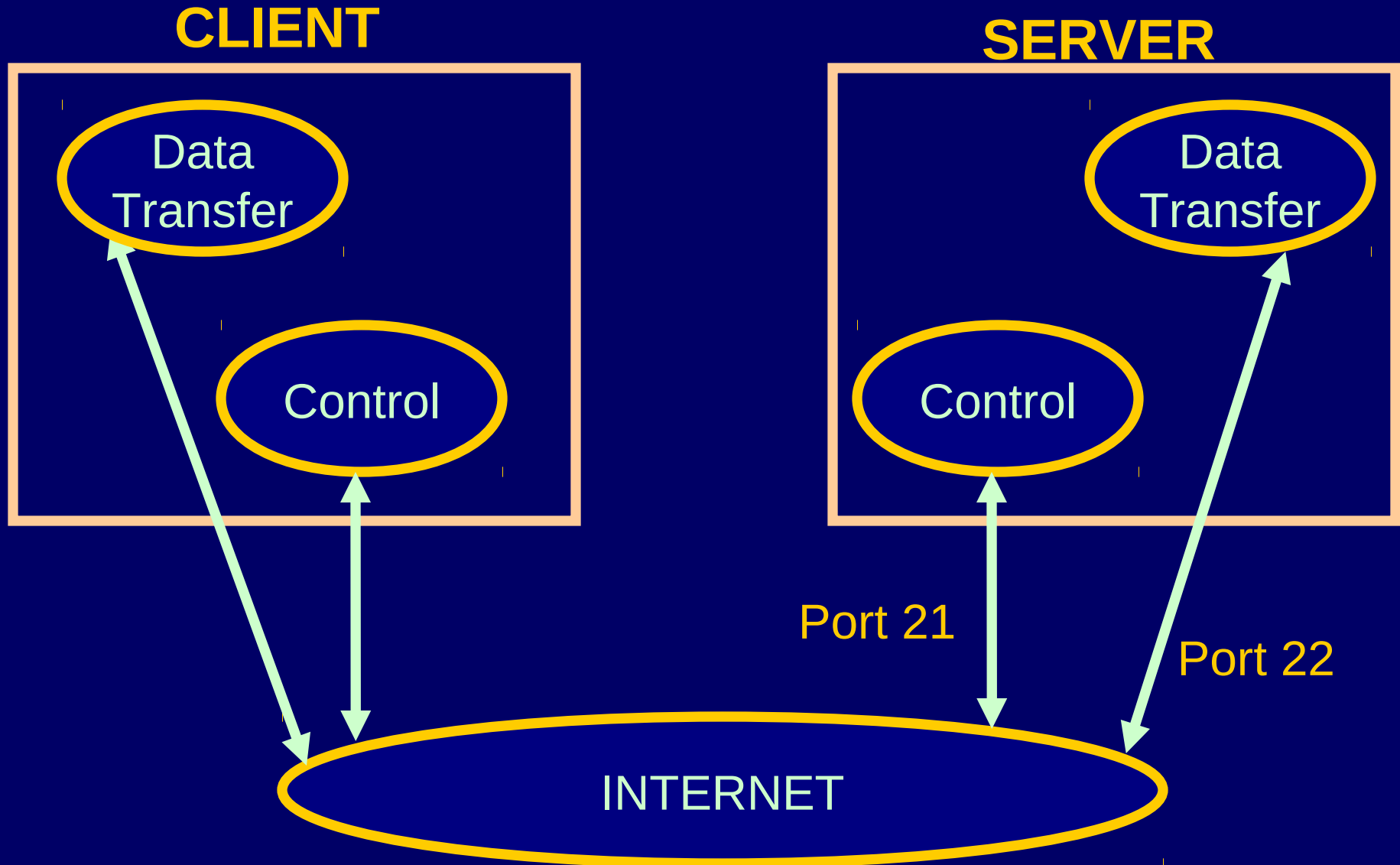


Anonymous FTP

- A special form of FTP which has become very popular.
 - Does not require a person to know login name and password.
 - In place of login name, type the word anonymous.
 - In place of password, type the email address.
- Huge amount of resources are available in anonymous FTP sites.



FTP Process Model





How FTP Works?

- **Two connections established:**
 - **A control connection over port 21, that remains all through a session.**
 - **A temporary port number, used for every file being transferred.**
 - **New connection established for every file transfer.**



Typical FTP Commands

- **put, mput**
 - Send one (or more) file(s).
- **get, mget**
 - Receive one (or more) file(s).
- **hash**
 - Indicates progress of file transfer.
- **prompt**
 - Prompts user before actual transfer of each file.
- **bin, asc**
 - Selects binary or ascii mode of transfer.
- **exit**
 - Terminates connection with ftp server.



Sample FTP Sessions

```
SUNNY>> ftp cseindigo
Connected to cseindigo.
220 cseindigo FTP server ready.
Name (cseindigo:isg): isg
331 Password required for isg.
Password:
230 User isg logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```



Sample FTP Sessions (contd.)

```
ftp> dir
```

```
200 PORT command successful.
```

```
150 Opening ASCII mode data connection for '/bin/lis'.
```

```
total 97
```

drwxr-xr-x	2	isg	faculty	512	Oct 8	1995	PROJECTS
-rw-r--r--	1	isg	faculty	7693	Mar 31	1995	lan-cse.fig
-rw-r--r--	1	isg	faculty	6578	Apr 26	1995	lan-cse.fig
-rw-r--r--	1	isg	faculty	6721	Jul 12	1995	newlan.fig

```
226 Transfer complete.
```



Sample FTP Sessions (contd.)

```
ftp> get lan-cse.fig
```

```
200 PORT command successful.
```

```
150 Opening BINARY mode data connection for 'lan-cse.fig'  
(6721 bytes).
```

```
226 Transfer complete.
```

```
6721 bytes received in 0.0389 secs (1.7e+02 Kbytes/sec)
```

```
ftp> quit
```

```
221 Goodbye.
```

```
SUNNY>>
```



End of Lecture 9