

# 文件传输

理论课程

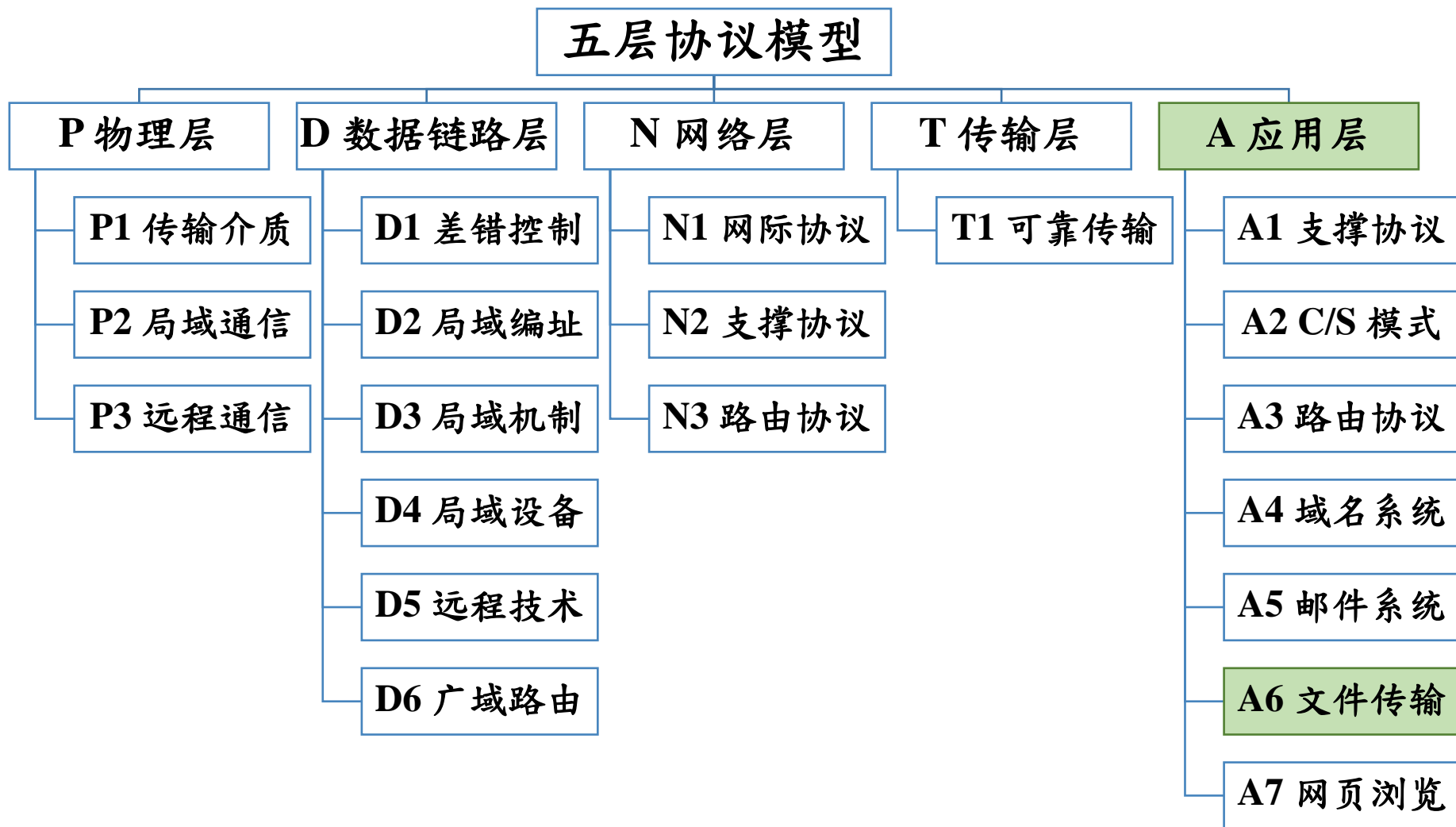


廈門大學  
XIAMEN UNIVERSITY



信息学院 黃 燁  
(国家示范性软件学院) 博士, 副教授  
School of Informatics Dr. Wei Huang

# 知识框架



# 主要内容

- **FTP**

- 工作原理与通信模式
- 主动和被动工作模式

# 对应课本章节

- **PART I Introduction And Internet Applications**
  - **Chapter 4 Traditional Internet Applications**
    - **4.10~4.11 File Transfer Protocol (FTP); FTP Communication Paradigm**

# 内容纲要

1	FTP协议
2	TFTP协议
3	NFS协议
4	SMB协议
5	Telnet协议

# FTP (File Transfer Protocol)

- 文件传送协议 FTP (File Transfer Protocol)
  - 提供交互式的访问，允许客户指明文件的类型与格式，并允许文件具有存取权限。
  - 屏蔽了各计算机系统的细节，因而适合于在异构网络中任意计算机之间传送文件。
  - RFC 959 很早就成为了因特网的正式标准。
- 文件传送并非很简单的问题
  - 众多的计算机厂商研制出的文件系统数百种，且差别很大。

# FTP 特点

- **FTP 使用基于流的客户服务器方式。**
  - 一个 FTP 服务器进程可同时为多个客户进程提供服务。
  - FTP 的服务器进程由两大部分组成：
    - 一个主进程，负责接受新的请求；
    - 另外有若干个从属进程，负责处理单个请求。

# FTP 两个不同的端口号

- FTP 使用2个端口号，数据连接与控制连接不会混乱。
  - 传输文件还可利用控制连接（如：客户发送请求终止传输）
  - 端口21：控制链接
    - 当客户软件向服务器软件发出建立连接请求时，寻找连接服务器软件的21号端口
    - 同时还告诉服务器软件自己的另一端口号，用于建立数据传送连接。
  - 端口20：数据连接
    - 接着，服务器软件用自己传送数据的20号端口与客户软件所提供的端口号码建立数据传送连接。



# FTP两个连接

- 控制连接

- 在整个会话期间一直保持打开，客户发出的传送请求通过控制连接发送给服务器端的控制进程，但不用来传送文件。

- 数据连接

- 实际用于传输文件。服务器端的控制进程在接收到 FTP 客户发送来的文件传输请求后创建“数据传送进程”和“数据连接”，用来连接客户端和服务器的数据传送进程。
- 数据传送进程实际完成文件的传送，在传送完毕后关闭“数据传送连接”并结束运行。

# FTP文件类型

- **ASCII 文件**

- 这是用于传输文本文件的默认格式。NVT ASCII编码。

- **EBCDIC文件（ 扩充的二—十进制交换码 ）**

- **Extended Binary-Coded Decimal Interchange Code.**

- **Image文件**

- 这是用于传输二进制文件的默认格式。文件被发送为连续的比特流，没有任何解释或编码。

- 这主要是用来传输二进制文件，如编译过的程序。

# 数据结构

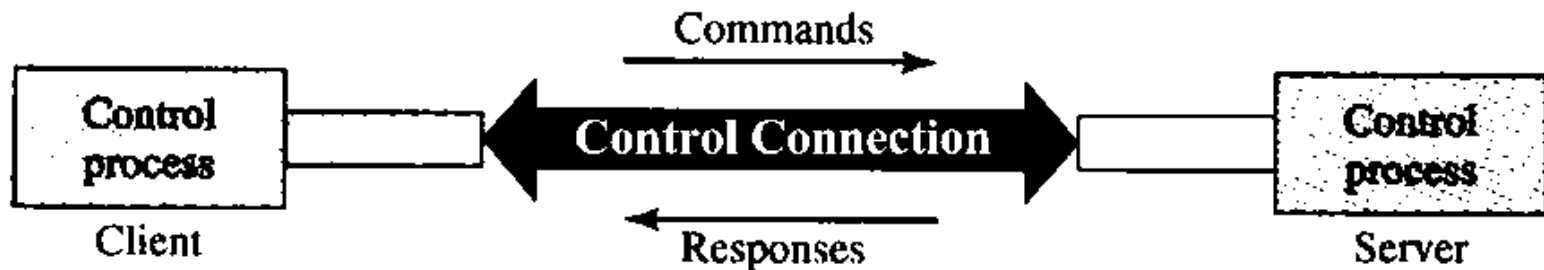
- 文件结构（默认）
  - 它是一个连续的字节流。
- 记录结构
  - 文件划分为记录。只用于文本文件。
- 页面结构
  - 文件分为两页，每一页有一个页和一个页头。
  - 该页可以随机或顺序存储或访问。

# 传输模式

- 流模式（默认）
  - 数据传送从FTP的TCP作为连续的字节流。
- 分块模式
  - 数据可以按分块从FTP到TCP分发。每块前有3字节的头。
- 压缩模式
  - 如果文件大，可以压缩数据。所使用的压缩方法通常是游程长度编码。在二进制文件中，空字符通常被压缩。

# 命令行处理

- FTP使用控制连接在客户机控制进程和服务端控制进程之间建立通信。



# FTP 命令

- 访问命令：这些命令让用户访问远程系统。

Table 20.1 Access commands

Command	Argument(s)	Description
USER	User id	User information
PASS	User password	Password
ACCT	Account to be charged	Account information
REIN		Reinitialize
QUIT		Log out of the system
ABOR		Abort the previous command

- 文件管理命令

Table 20.2 File management commands

Command	Argument(s)	Description
CWD	Directory name	Change to another directory
CDUP		Change to the parent directory
DELE	File name	Delete a file
LIST	Directory name	List subdirectories or files
NLIST	Directory name	List the names of subdirectories or files without other attributes
MKD	Directory name	Create a new directory

Table 20.2 File management commands (continued)

Command	Argument(s)	Description
PWD		Display name of current directory
RMD	Directory name	Delete a directory
RNFR	File name (old file name)	Identify a file to be renamed
RNTO	File name (new file name)	Rename the file
SMNT	File system name	Mount a file system

# FTP 命令

- 数据格式命令

Table 20.3 Data formatting commands

Command	Argument(s)	Description
<b>TYPE</b>	A (ASCII), E (EBCDIC), I (Image), N (Nonprint), or T (TELNET)	Define the file type and if necessary the print format
<b>STRU</b>	F (File), R (Record), or P (Page)	Define the organization of the data
<b>MODE</b>	S (Stream), B (Block), or C (Compressed)	Define the transmission mode

- 端口定义命令

— 用户定义数据连接的客户端的端口号

Table 20.4 Port defining commands

Command	Argument(s)	Description
<b>PORT</b>	6-digit identifier	Client chooses a port
<b>PASV</b>		Server chooses a port

# FTP 命令

## • 文件传输命令

Table 20.5 File transfer commands

Command	Argument(s)	Description
<b>RETR</b>	File name(s)	Retrieve files; file(s) are transferred from server to the client
<b>STOR</b>	File name(s)	Store files; file(s) are transferred from the client to the server
<b>APPE</b>	File name(s)	Similar to STOR except if the file exists, data must be appended to it
<b>STOU</b>	File name(s)	Same as STOR except that the file name will be unique in the directory; however, the existing file should not be overwritten
<b>ALLO</b>	File name(s)	Allocate storage space for the files at the server
<b>REST</b>	File name(s)	Position the file marker at a specified data point
<b>STAT</b>	File name(s)	Return the status of files



# FTP 响应

## • 响应代码

Table 20.7 Responses

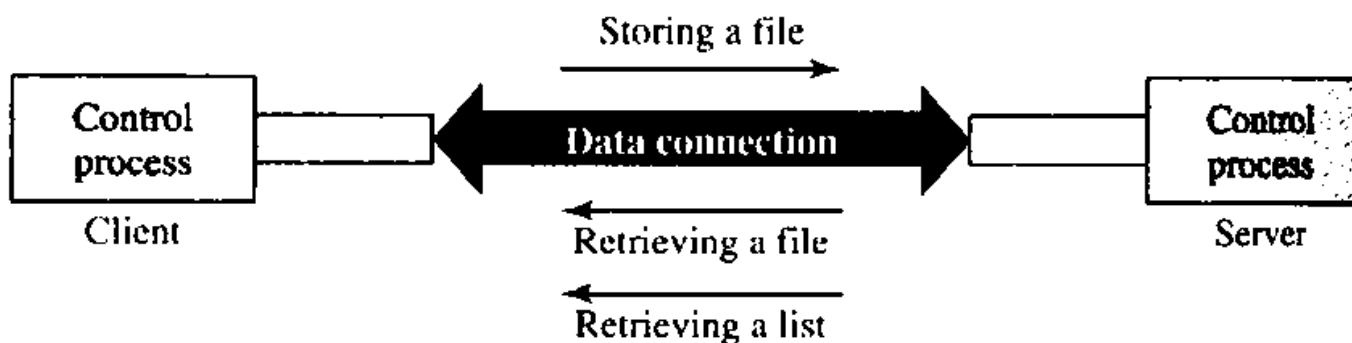
Code	Description
<b>Positive Preliminary Reply</b>	
120	Service will be ready shortly
125	Data connection open; data transfer will start shortly
150	File status is OK; data connection will be open shortly
<b>Positive Completion Reply</b>	
200	Command OK
211	System status or help reply
212	Directory status
213	File status
214	Help message
215	Naming the system type (operating system)
220	Service ready
221	Service closing
225	Data connection open
226	Closing data connection
227	Entering passive mode; server sends its IP address and port number
230	User login OK
250	Request file action OK

Table 20.7 Responses (continued)

Code	Description
<b>Positive Intermediate Reply</b>	
331	User name OK; password is needed
332	Need account for logging
350	The file action is pending; more information needed
<b>Transient Negative Completion Reply</b>	
425	Cannot open data connection
426	Connection closed; transfer aborted
450	File action not taken; file not available
451	Action aborted; local error
452	Action aborted; insufficient storage
<b>Permanent Negative Completion Reply</b>	
500	Syntax error; unrecognized command
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command parameter not implemented
530	User not logged in
532	Need account for storing file
550	Action is not done; file unavailable
552	Requested action aborted; exceeded storage allocation
553	Requested action not taken; file name not allowed

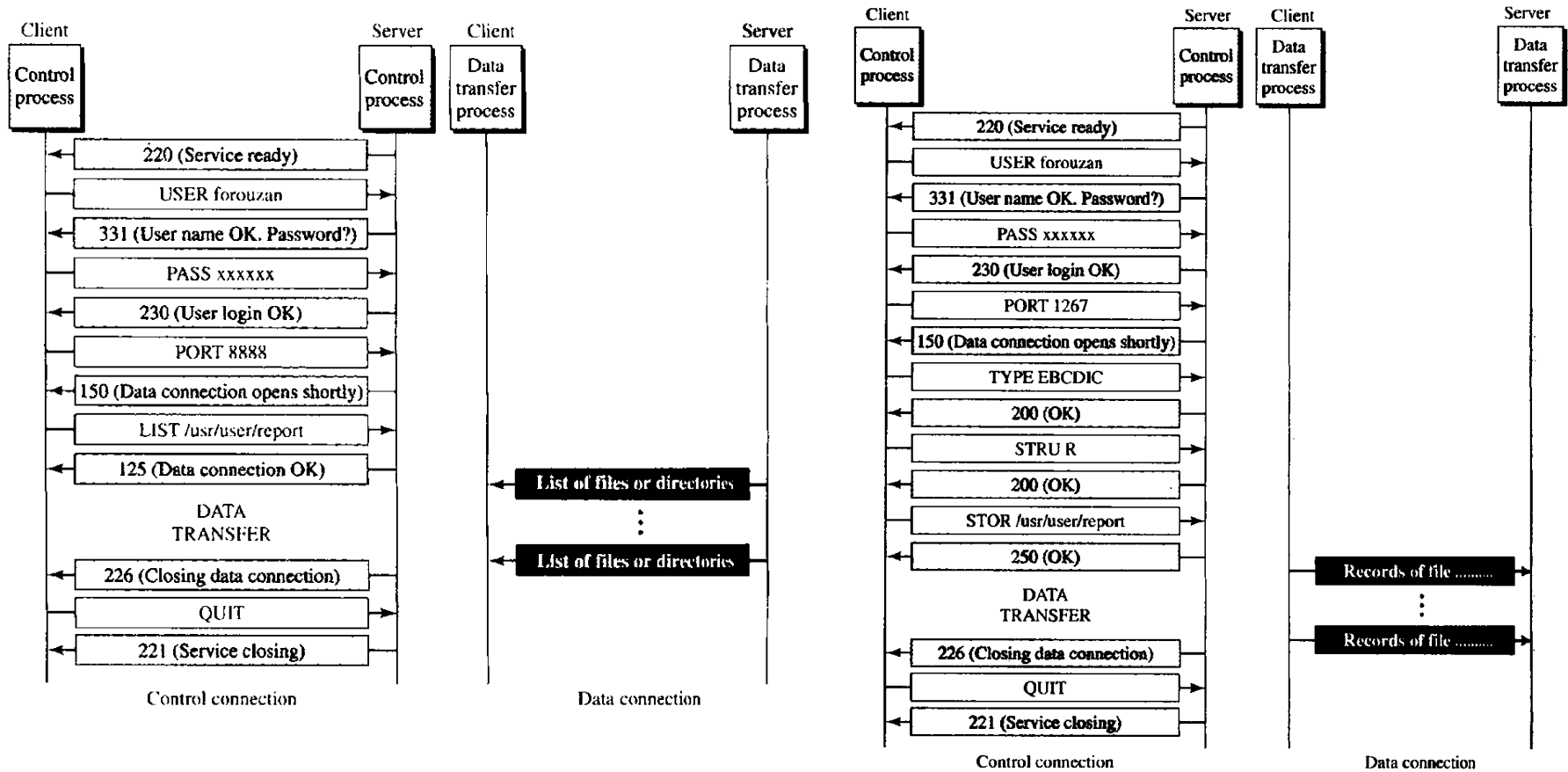
# 文件传输

- 在控制连接上设置的命令的控制下，在数据连接上发生文件传输。



# 范例

## • CS交互范例



# 内容纲要

1	FTP协议
2	TFTP协议
3	NFS协议
4	SMB协议
5	Telnet协议

# 简单文件传输协议 ( TFTP )

- **Trivial File Transfer Protocol ( TFTP )**
  - TFTP 使用 UDP 服务的端口 69.
- TFTP 是一种简化的 TCP/IP 文件传输协议。
- TFTP 只限于简单文件传输操作，它不提供权限控制，也不支持客户与服务器之间复杂的交互过程，没有庞大的命令集，没有列目录的功能，不能鉴别用户身份。因此 TFTP 软件比 FTP 软件小的多。

# TFTP

- TFTP 是一个很小的文件传送协议。
- TFTP 使用客户服务器方式和使用 UDP 数据报，因此 TFTP 需要有自己的差错改正措施。
- TFTP 只支持文件传输而不支持交互。
- TFTP 没有一个庞大的命令集，没有列目录的功能，也不能对用户进行身份鉴别。

# TFTP 的主要特点

- 每次传送的数据 PDU 中有 512 字节的数据，但最后一次可不足 512 字节。
- 数据 PDU 也称为文件块(block)，每个块按序编号，从 1 开始。
- 支持 ASCII 码或二进制传送。
- 可对文件进行读或写。
- 使用很简单的首部。

# TFTP 的工作类似停止等待协议

- 发送完一个文件块后就等待对方的确认，确认时应指明所确认的块编号。
- 发完数据后在规定时间内收不到确认就要重发数据 PDU。
- 发送确认 PDU 的一方若在规定时间内收不到下一个文件块，也要重发确认 PDU。这样就可保证文件的传送不致因某一个数据报的丢失而告失败。



# TFTP 的工作类似停止等待协议

- 在一开始工作时。TFTP 客户进程发送一个读请求 PDU 或写请求 PDU 给 TFTP 服务器进程，其熟知端口号码为 69。
- TFTP 服务器进程要选择一个新的端口和 TFTP 客户进程进行通信。
- 若文件长度恰为 512 字节的整数倍，则文件传送完毕后还必须最后发送一个只含首部而无数据的数据 PDU。
- 若文件长度不是 512 字节的整数倍，则最后传送数据 PDU 的数据字段一定不满 512 字节，这正好可作为文件结束的标志。

# 内容纲要

	1	FTP协议
	2	TFTP协议
	3	NFS协议
	4	SMB协议
	5	Telnet协议

# 网络文件系统（ NFS ）

- 网络文件系统（ Network File System ， NFS ）
- NFS只是一种文件系统，本身没有传输功能，是基于RPC协议实现的，才能达到两个Linux系统之间的文件目录共享；

# Network File System

- NFS 允许应用进程打开一个远程文件，并能在该文件的某一个特定的位置上开始读写数据。
- NFS 可使用户只复制一个大文件中的一个很小的片段，而不需要复制整个大文件。
- 例：计算机 A 的 NFS 客户软件，把要添加的数据和在文件后面写数据的请求一起发送到计算机 B 的 NFS 服务器。NFS 服务器更新文件后返回应答信息。
- 在网络上传送的只是少量的修改数据。

# 内容纲要

	1	FTP协议
	2	TFTP协议
	3	NFS协议
	4	SMB协议
	5	Telnet协议

# 简单文件共享：SMB协议

- Server Message Block
- 基于TCP-NETBIOS下的，一般端口为139、445。
  - NetBIOS（网络基本输入/输出系统协议）协议是由IBM公司开发，主要用于数十台计算机的小型局域网。
  - NetBIOS协议是一种在局域网上的程序可以使用的API，为程序提供了请求低级服务的统一的命令集
  - 几乎所有的局域网都是在NetBIOS协议的基础上工作的。

# 远程终端协议 Telnet

- Telnet 是一个简单的远程终端协议。
- 用户用 Telnet 就可在其所在地通过 TCP 连接注册（即登录）到远程的另一个主机上（使用主机名或 IP 地址）。
- Telnet 能将用户的击键传到远程主机，同时也能将远程主机的输出通过 TCP 连接返回到用户屏幕。这种服务是透明的，因为用户感觉到好像键盘和显示器是直接连在远程主机上。

# 内容纲要

	1	FTP协议
	2	TFTP协议
	3	NFS协议
	4	SMB协议
	5	Telnet协议

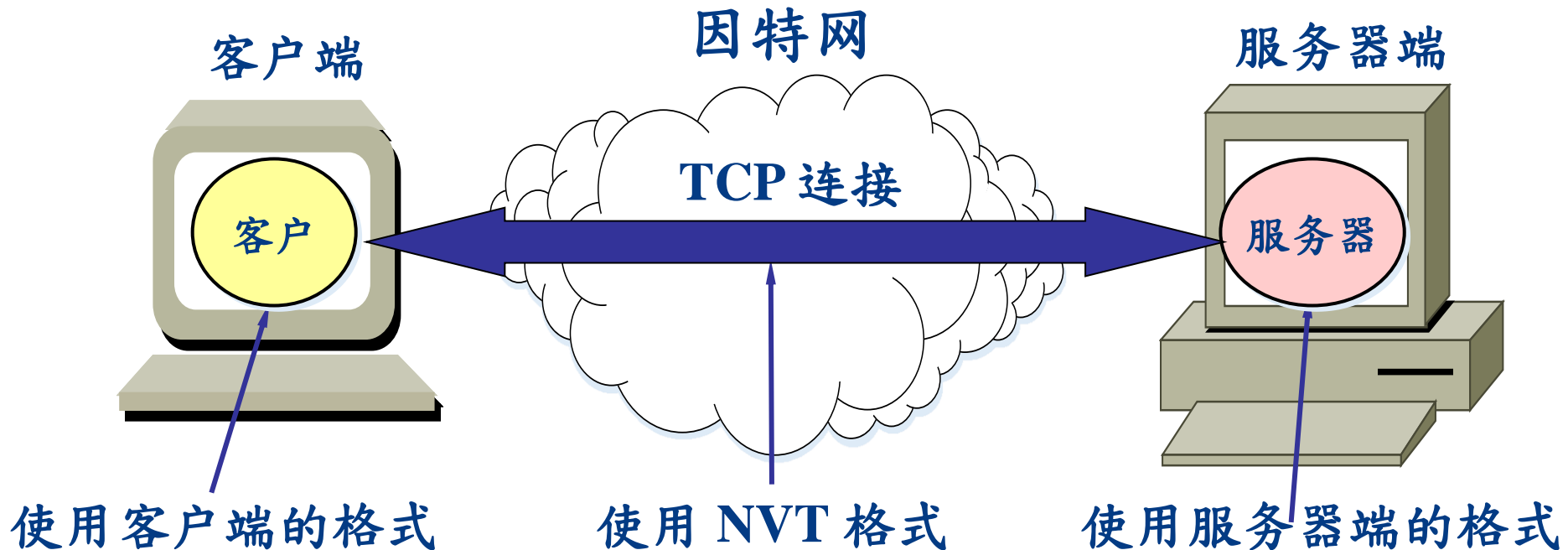


# 客户-服务器方式

- 现在由于 PC 的功能越来越强，用户已较少使用 Telnet 了。（但在 CUI 界面系统仍使用）
- Telnet 也使用客户-服务器方式。在本地系统运行 Telnet 客户进程，而在远程主机则运行 Telnet 服务器进程。
- 和 FTP 的情况相似，服务器中的主进程等待新的请求，并产生从属进程来处理每一个连接。

# Telnet 使用NVT 格式

- 网络虚拟终端 NVT



# 网络虚拟终端 NVT 格式

- 客户软件把用户的击键和命令转换成 NVT 格式，并送交服务器。
- 服务器软件把收到的数据和命令，从 NVT 格式转换成远程系统所需的格式。
- 向用户返回数据时，服务器把远程系统的格式转换为 NVT 格式，本地客户再从 NVT 格式转换到本地系统所需的格式。

谢谢观看



廈門大學  
XIAMEN UNIVERSITY



信息学院 黄 焯  
(国家示范性软件学院) 博士, 副教授  
School of Informatics Dr. Wei Huang