



传输层

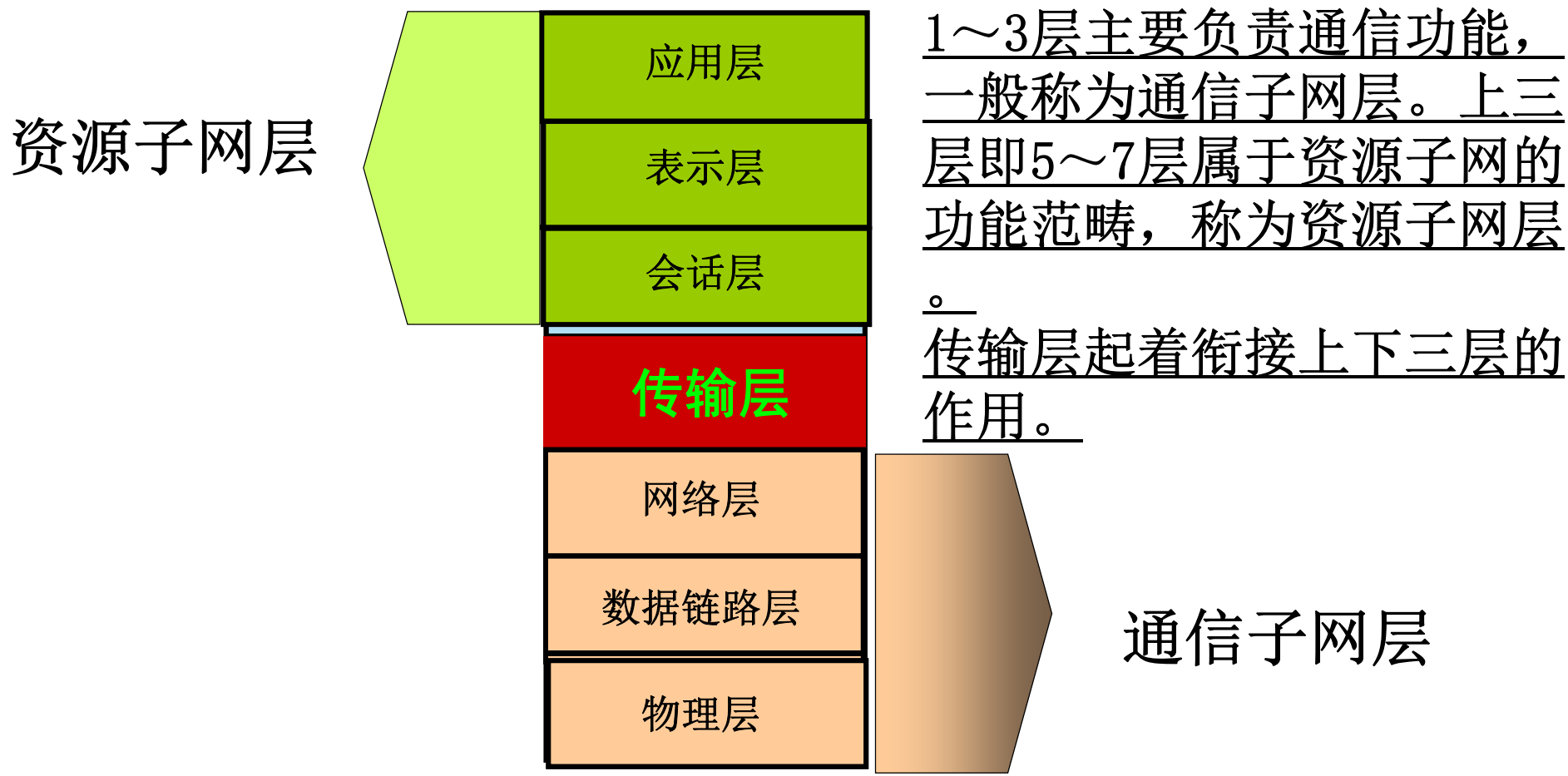


传输层的任务

将数据从进程传递到进程。实现进程之间的数据传递。

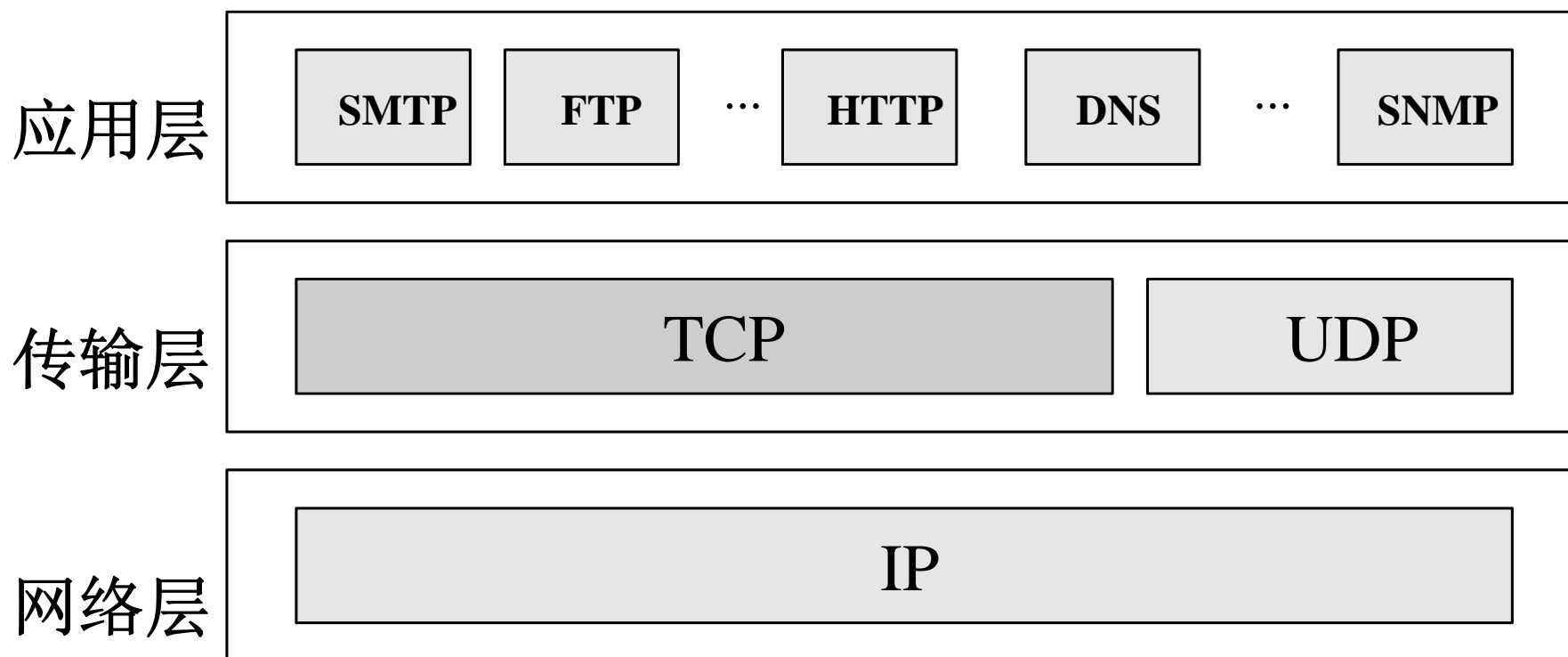


传输层在OSI协议层次结构中的位置





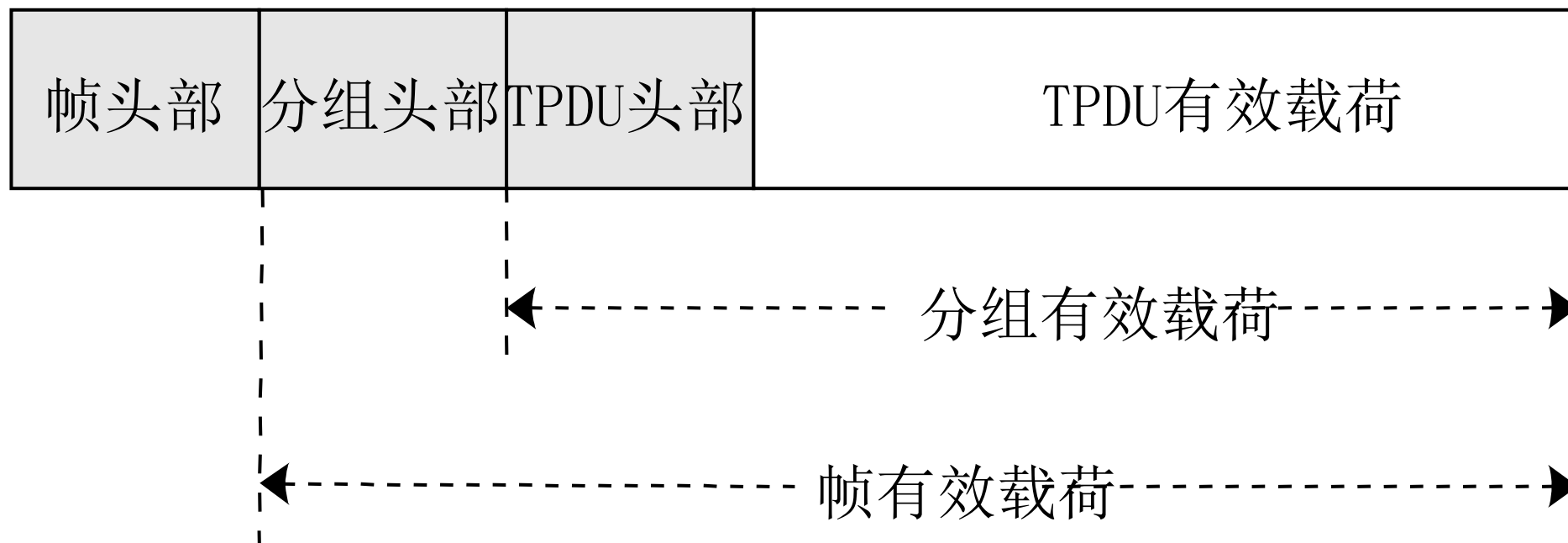
TCP/IP分层中的传输层协议结构





传输协议数据单元

- ✓ 传输层之间传输的报文叫做传输协议数据单元（Transport Protocol Unit, TPDU）；
- ✓ TPDU有效载荷是应用层的数据。





单机系统中的进程通信方法

- 进程
- 从进程的观点看，操作系统的核心则是控制和协调这些进程的运行，解决进程之间的通信。



网络环境与单机系统内部的进程通信的区别

- ✓ 网络中主机的高度自治性;
- ✓ 不是在同一个主机系统之中, 没有一个统一的高层进行控制与管理;
- ✓ 网络中一台主机对其他主机的
 - 活动状态;
 - 位于其他主机系统中的各个进程状态;
 - 这些进程什么时间参与网络活动;
 - 希望与网络中哪一台主机的什么进程通信;一概无从知道。



网络环境中进程通信需要解决

- ✓ 进程标识（三元组）
- ✓ 进程间相互作用的模式



网络环境中 进程标识（三元组）

网络环境中完整的进程标识应该是：

- 主机地址
- 进程端口
- 进程使用的传输层协议（TCP/UDP）

标识网络中的一个Web服务器：（IP, 80, TCP）



网络环境中 进程标识（三元组）

标识网络中的一个Web服务器：(IP, 80, TCP)

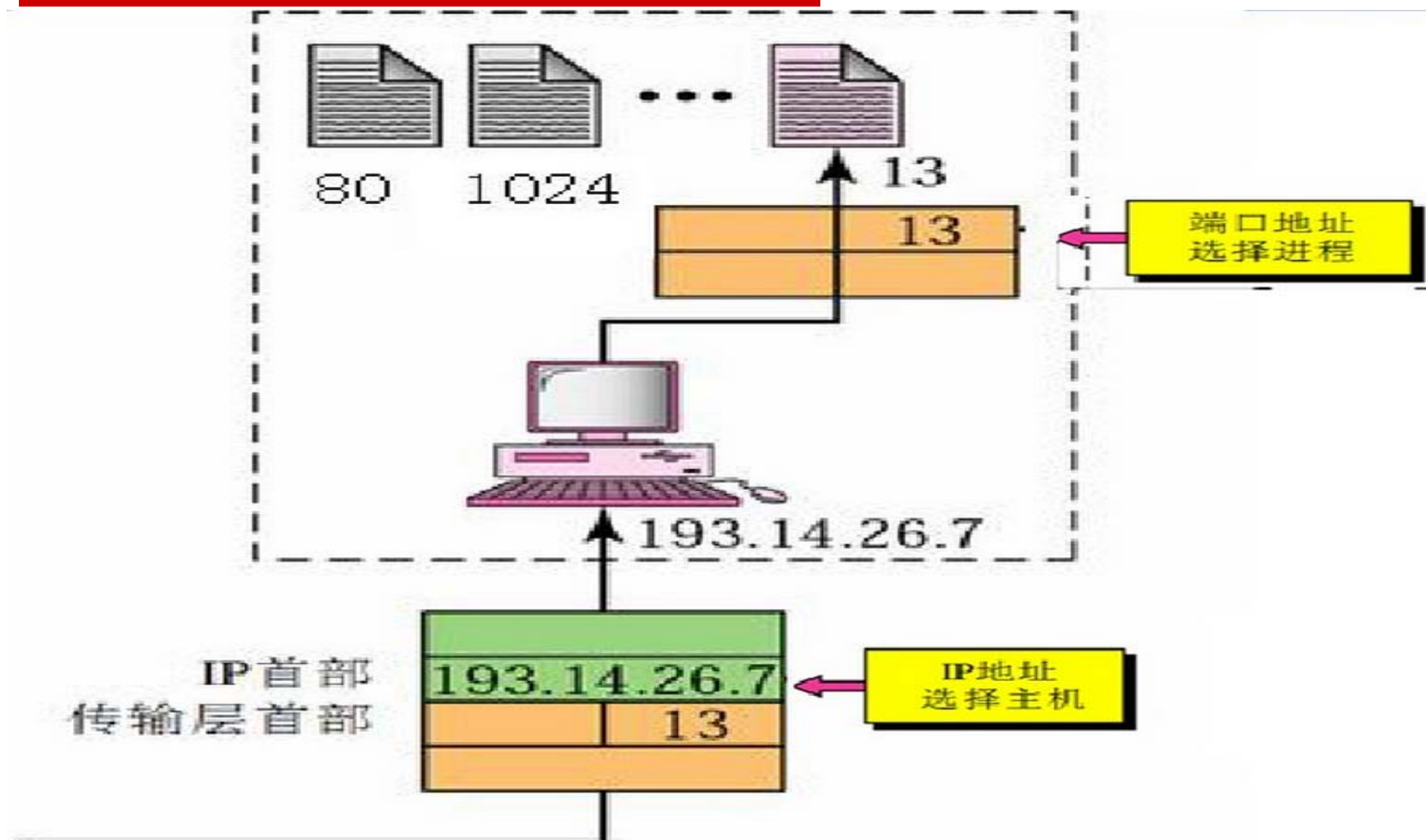


进程端口

- 从0 到255 被规定作为公共应用服务的端口，如WWW、FTP、DNS 和电子邮件服务等，又被称为著名端口 (Well-known ports)。这些端口所对应的服务是固定、公开的，因此可以通过访问这些端口来访问对应的服务。
- 从256 到1023 的端口，被保留用作商业性的应用开发，如一些网络设备厂商专用协议的通信端口等。
- 1023 以上端口未做限定，即作为自由端口，以本地方式进行分配。



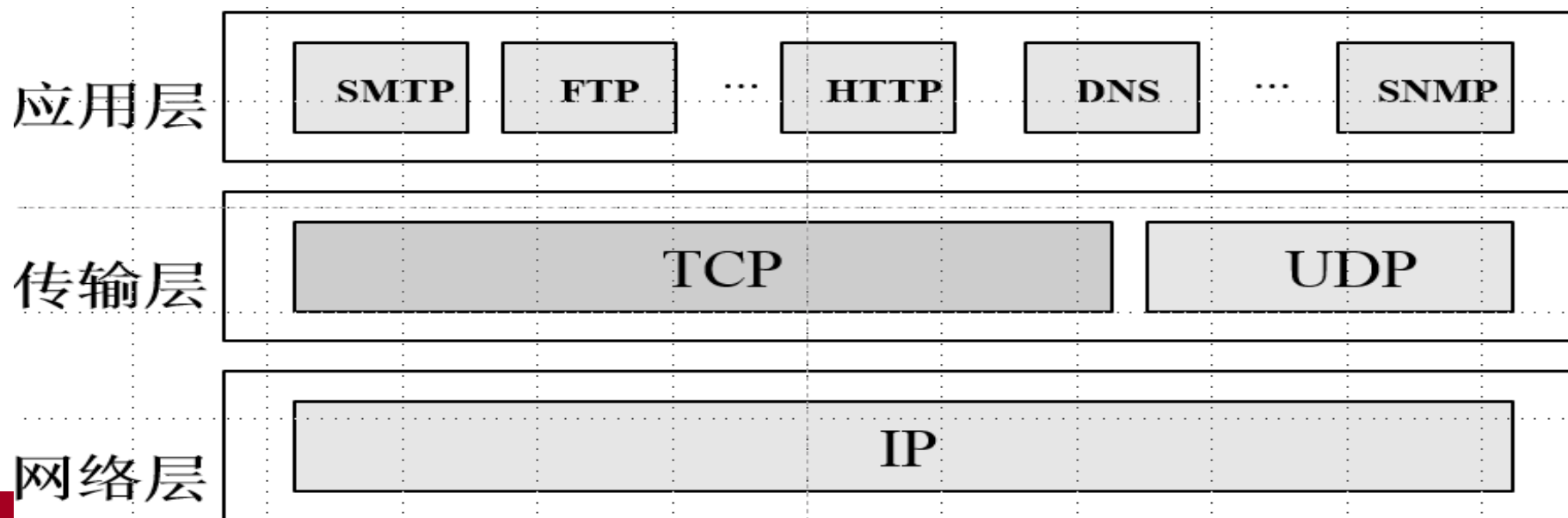
进程端口





进程使用的传输层协议（多重协议的识别）

- ✓ TCP/IP协议族的传输层有TCP协议和UDP协议；
- ✓ 应用层进程会选择传输层TCP, UDP中的一个进行数据传输；
- ✓ 基于TCP传输的端口在0-65535之间，基于UDP传输的进程的端口在0-65535.





一对通信的进程的标识（五元组）

- ✓ 双方使用的传输层协议
- ✓ A所在的主机地址
- ✓ A进程端口
- ✓ B所在的主机地址
- ✓ B进程端口



进程间相互作用模式：Client/Server模型

- 客 户——一次进程通信中发起的一方；
- 服务器——接受进程通信的请求，提供服务的一方；
- 每一次通信由客户进程随机启动；
- 服务器进程处于等待状态，及时响应客户服务请求。



为什么要采用客户机/服务器模型？

- ✓ 网络资源分布的不均匀性
- ✓ 网络资源分布的不均匀性表现在硬件、软件和数据等三个方面；
- ✓ 网络资源分布的不均匀性是客观存在的，同时也是网络应用系统设计者的设计思想的体现；
- ✓ “资源共享”就是因为网络不同结点之间在硬件配置、计算能力、存储能力，以及数据分布等方面存在着差距与不均匀性；
- ✓ 能力强、资源丰富的充当服务器，能力弱或需要某种资源的成为客户。



服务器处理并发请求的基本方案：

- ✓ 并发服务器；
- ✓ 重复服务器。



并发服务器（concurrent server）

- 并发服务器的核心一个守护程序（**daemon**），守护程序在系统启动时启动，在没有客户服务请求到达时，并发服务器处于等待状态；
- 客户机的服务请求到达时，服务器根据客户的服务请求的进程号激活相应子进程，服务器回到等待状态；
- 并发服务器叫做主服务器（**master**），服务器叫做从服务器（**slave**）；
- 并发服务器拥有一个全网唯一的进程地址，网络中的客户进程可以根据该地址，向服务器提出服务请求。



重复服务器 (iterative server)

- 通过设置一个请求队列来存储客户机的服务请求；
- 服务器采用先来先服务的原则来顺序处理客户机的服务请求。



作业

- 1. 简述传输层的基本功能。
- 2. 简述如何在互联网中唯一标识一个参与网络通信的进程。
- 3. 简述网络中进行通信的基本模型。