

1、为什么要网络编程？ ----丰富的网络资源

2、什么是网络编程？

3、网络编程中的基本概念

1. 发送端和接收端

2. 请求和响应

3. 客户端和服务端

4. 常见的客户端服务端模型

1、为什么要网络编程？ ----丰富的网络资源

用户在浏览器中，打开在线视频网站，如优酷看视频，实质是通过网络，获取到网络上的一个视频资源



与本地打开视频文件类似，只是视频文件这个资源的来源是网络。

相比本地资源来说，网络提供了更为丰富的网络资源：

网络资源

视频资源

图片资源

文本资源

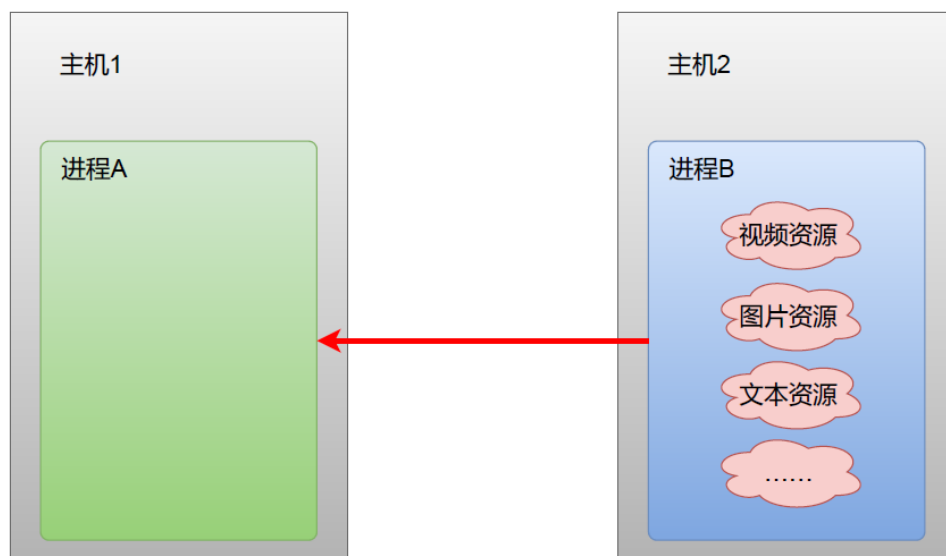
文本资源

所谓的网络资源，其实就是在网络中可以获取的各种数据资源。

而所有的网络资源，都是通过**网络编程**来进行数据传输的。

2、什么是网络编程？

网络编程，指网络上的主机，通过不同的进程，以编程的方式实现网络通信（或称为网络数据传输）。



只要满足进程不同就行；所以即便是同一个主机，只要是不同进程，基于网络来传输数据，也属于网络编程。

特殊的，对于开发来说，在条件有限的情况下，一般也都是在同一个主机中运行多个进程来完成网络编程。

但是，我们一定要明确，我们的目的是提供网络上不同主机，基于网络来传输数据资源：

- 进程A：编程来获取网络资源
- 进程B：编程来提供网络资源

3、网络编程中的基本概念

1. 发送端和接收端

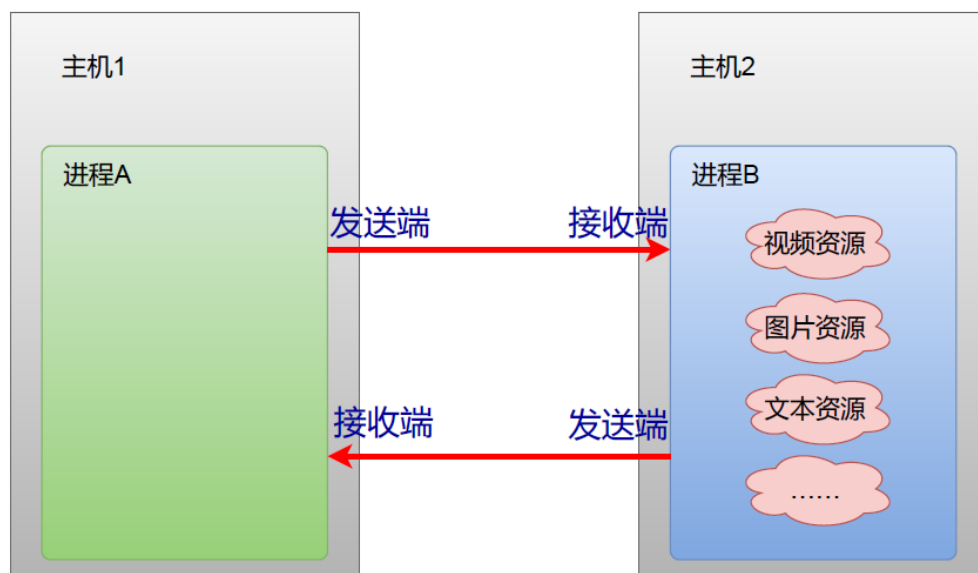
在一次网络数据传输时：

发送端：数据的发送方进程，称为发送端。发送端主机即网络通信中的源主机。

接收端：数据的接收方进程，称为接收端。接收端主机即网络通信中的目的主机。

收发端：发送端和接收端两端，也简称为收发端。

注意：发送端和接收端只是相对的，只是一次网络数据传输产生数据流向后的概念。

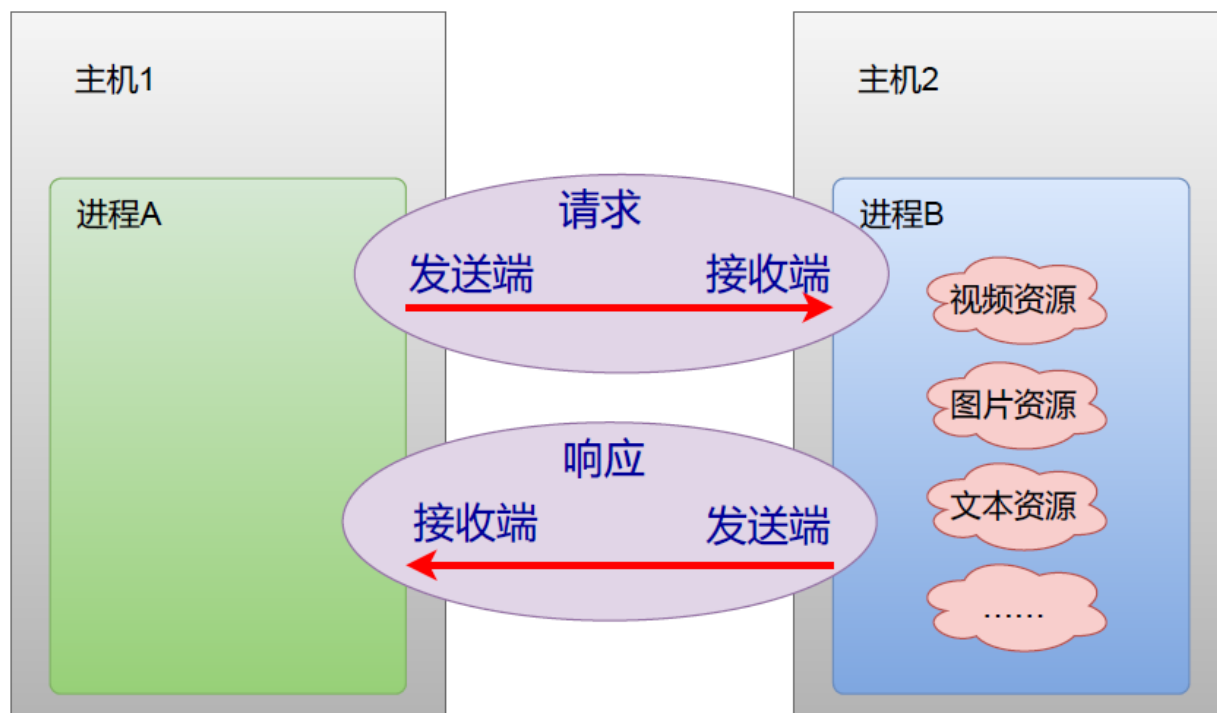


2. 请求和响应

一般来说，获取一个网络资源，涉及到**两次网络数据传输**：

第一次：**请求**数据的发送

第二次：**响应**数据的发送。



3. 客户端和服务端

服务端：在常见的网络数据传输场景下，把**提供服务**的一方进程，称为**服务端**，可以提供对外服务。

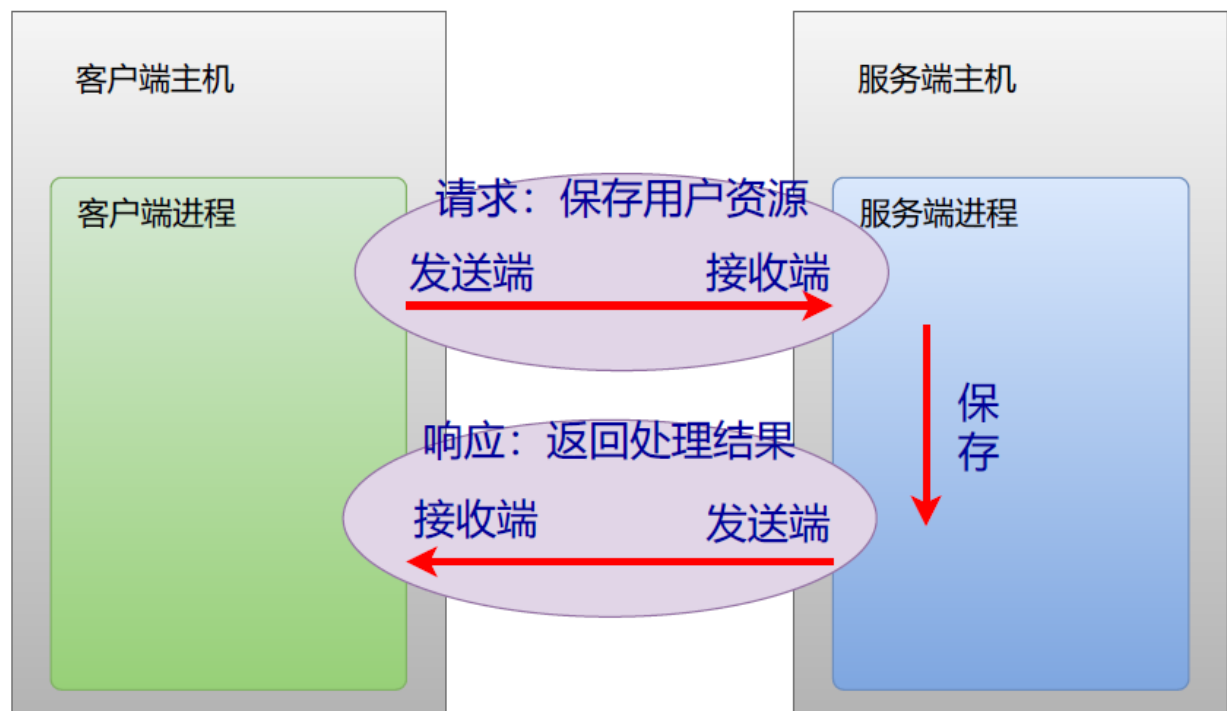
客户端：**获取服务**的一方进程，称为客户端。

对于服务来说，一般是提供：

- **客户端获取服务资源**



- 客户端保存资源在服务端



4. 常见的客户端服务端模型

最常见的场景，客户端是指给用户使用的程序，服务端是提供用户服务的程序：

1. 客户端先发送请求到服务端
2. 服务端根据请求数据，执行相应的业务处理
3. 服务端返回响应：发送业务处理结果

4. 客户端根据响应数据，展示处理结果（展示获取的资源，或提示保存资源的处理结果）

