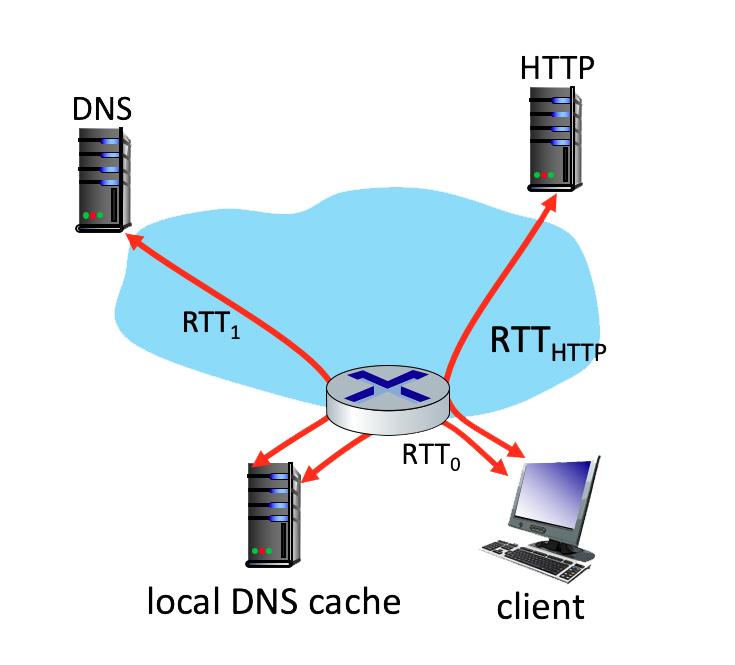
**Exercise #1**



Suppose within your Web browser you click on a link to obtain a Web page. The IP address for the associated URL is not cached in your local host, so a DNS lookup is necessary to obtain the IP address. Suppose that two DNS servers are visited before your host receives the IP address from DNS. The first DNS server visited is the local DNS cache, with an RTT delay of RTT0 = 2 msecs. The second DNS server contacted has an RTT of 26 msecs. Initially, let's suppose that the Web page associated with the link contains exactly one object, consisting of a small amount of HTML text. Suppose the RTT between the local host and the Web server containing the object is RTTHTTP = 83 msecs.

**Questions**

1. Assuming zero transmission time for the HTML object, how much time (in msec) elapses from when the client clicks on the link until the client receives the object?  
  
2. Now suppose the HTML object references 4 very small objects on the same server. Neglecting transmission times, how much time (in msec) elapses from when the client clicks on the link until the base object and all 4 additional objects are received from web server at the client, assuming non-persistent HTTP and no parallel TCP connections?  
  
3. Suppose the HTML object references 4 very small objects on the same server, but assume that the client is configured to support a maximum of 5 parallel TCP connections, with non-persistent HTTP.  
  
4. Suppose the HTML object references 4 very small objects on the same server, but assume that the client is configured to support a maximum of 5 parallel TCP connections, with persistent HTTP.  
  
5. What's the fastest method we've explored: Nonpersistent-serial, Nonpersistent-parallel, or Persistent-parallel?

1. Assuming zero transmission time for the HTML object, how much time (in msec) elapses from when the client clicks on the link until the client receives the object?

**Ans:**

RTT0 + RTT1 + 2\*RTTHTTP

1. RTT0 + RTT1 + 2\*RTTHTTP + 2\*4\*RTTHTTP
2. 28 + 166 + 166 = 360 msec.
3. 28 + 166 + 83 = 277 msec

**5.**