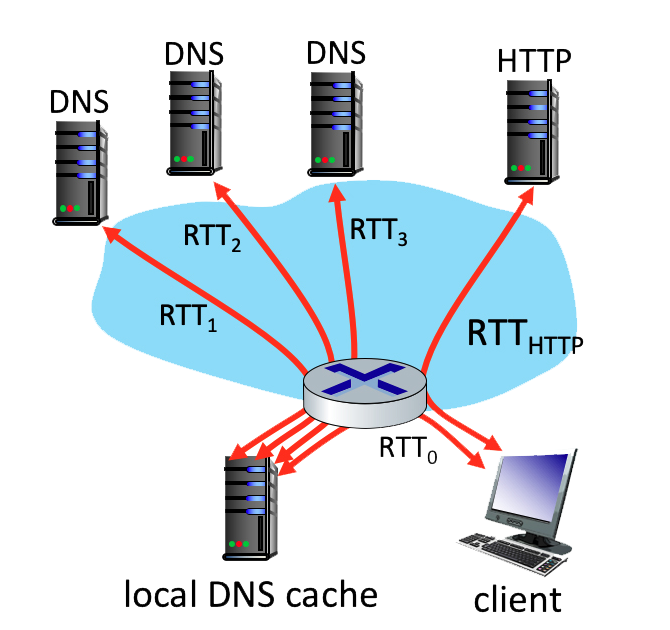
1. **Did you do the** [**HTTP**](http://www-net.cs.umass.edu/wireshark-labs/Wireshark_HTTP_v8.0.pdf) **Wireshark lab?**
   * 1. Yes
2. **Suppose you and four of your friends are trying to load the same webpage.**
   1. **You decide to open multiple parallel non-persistent HTTP connections. Will this help you load the webpage more quickly than your friends? Why or why not?**
      * + 1. Opening multiple connections will help load the webpage more quickly because it allows your computer to send more requests to the server and receive multiple things at once rather than having to wait for each item to load one after another
   2. **If all of your friends do the same thing and open the same number of parallel non-persistent HTTP connections as you, will your parallel connections still be beneficial? Why or why not?**
      * + 1. The parallel connections will still be beneficial because the server is made to send out many responses to requests all at once. If five people are opening multiple connections to the server, all will be able to make multiple requests for items on the page to load.
3. **What is the difference between non-persistent HTTP connections and persistent HTTP connections? Which one is better and why?**
   * 1. Non-persistent connections have to make a new TCP connection for each request that they want to send. Persistent uses the same connection and will save time because the round trip time will be lower. It can send requests back to back without having to open a new TCP connection to the server each time.
4. **  
   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 four 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 = 5 ms. The second, third and fourth DNS servers contacted have RTTs of 35, 30, and 1 ms, respectively. 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 = 6 ms.**
   1. **Assuming zero transmission time for the HTML object, how much time (in ms) elapses from when the client clicks on the link until the client receives the object?**
      * + 1. (5+35+30+1)+(2\*6) = 83ms
   2. **Now suppose the HTML object references 2 very small objects on the same server. Neglecting transmission times, how much time (in ms) elapses from when the client clicks on the link until the base object and all 2 additional objects are received from web server at the client, assuming non-persistent HTTP and no parallel TCP connections?**
      * + 1. (5+35+30+1)+(2\*6)\*3 = 107ms
   3. **Suppose the HTML object references 2 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.**
      * + 1. (5+35+30+1)+(2\*6)\*2 = 95ms
   4. **Suppose the HTML object references 2 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.**
      * + 1. (5+35+30+1)+(3\*6) = 89ms
   5. **What’s the fastest method we’ve explored: Nonpersistent-serial, Nonpersistent-parallel, or Persistent-parallel?**
      * + 1. Persistent-parallel
5. **How does SMTP mark the end of a message body? How about HTTP? Can HTTP use the same method as SMTP to mark the end of a message body? Explain.**
   * 1. SMTP uses a period to mark the end of the message body. Http uses a “content‐Length header field.” HTTP can’t use the period because it could have binary data whereas SMTP uses a 7 bit ASCII character to represent the period character.
6. **Suppose you can access the caches in the local DNS servers of your department. Can you propose a way to roughly determine the Web servers (outside your department) that are most popular among the users in your department? Explain.**
   * 1. By accessing the local DNS servers in the department, you can see which web servers are requested more often than others. This will cache in the local DNS and show which site are the most popular in the department.
7. **Suppose that your department has a local DNS server for all computers in the department. You are an ordinary user (i.e., not a network administrator). Can you determine if an external website was likely accessed recently from a computer in your department? Explain.**
   * 1. Through the command line, you can use dig followed by an external site to see the query time. This is stored in the local DNS cache. If the query time is low, then the site was recently accessed from a computer in the department.
8. **Suppose you join a BitTorrent torrent, but you do not want to upload any data to any other peers (free-riding). Can you receive a complete copy of a file that is shared by the swarm. Why or why not?**
   * 1. Possibly but probably not. Whenever you torrent something, you usually automaticcaly start slowly uploading the chunks of the files you already have so new people can start downloading it too. If you aren’t uploading, or seeding, your IP address can be flagged and you could be marked leecher, or someone who hits-and-runs, meaning that you selfishly downloaded files but didn’t give back to the community by uploading in return. As such, you may be blacklisted from torrent trackers and unable to torrent future files.