Hi ,  
Lets talk a little about human - computer address translator , the DNS (or by it's full name : Domain Name System) .  
Like you know we use sites names when we talk to google something we go to google.com , to view stuff regarding this site we go to coursera.org etc ... .  
Since it's not practical for computer to search where is the "coursera.org" site information is located it needs some other way to find the site , this way is called IP address and it looks something like 192.0.43.10 for IP4 and 2001:500:88:200::10 for IP6 . IP6 is still not widely used , and was created to allow more IP address then IP4 so lets concentrate on IP4 format. This format works like phone structure in which the first digits specify the country then next few the carrier and then the remain digits the actual person in the carrier network , much alike in IP the first part is network identifier and the second is the host (specific computer or phone) identifier in that network.   
So for computer it is much easier and quicker to search according IP address then according the "human" address . Actually many sites have few IP addresses that correspond to the same "human" address , thus allowing load balancing without any action (or even knowledge) of the users. Human mind on the other hand works much better remembering meaningful words instead of long seemingly random IP addresses.  
This gap is filled by the DNS , but before we'll dive into DNS details there is another little issue that we need to understand : Who is responsible for maintaining the sites data ? When you access [www.mail.yahoo.com](http://www.mail.yahoo.com) who is responsible for this site ? Who's responsibility is it to know how to translate the mail.yahoo.com name to appropriate IP name ? To know the answer for that question we need to start looking on the url and spit it to "worlds" backwards . So in our example we'll get : com , yahoo , mail , www , and we'll need to start questioning them . We'll start with the com part , well there is organization that is responsible for all domains that "end" with .com :  [Internet Assigned Numbers Authority](http://en.wikipedia.org/wiki/Internet_Assigned_Numbers_Authority) (IANA) (the top most domain can also belong to specific government , for example .uk belongs to United Kingdom) , so we'll start with them and ask them if they are responsible for our IP translation , they would say "no , but ... if you really want to know we suggest you talk to yahoo , since they bought this domain from us" , so we'll ask yahoo if they are responsible for our IP translation and they will answer "well , not quite , but ... our email department uses that domain so you should talk with them" , so we're going to ask them and we get the answer "yes , yes ... we are responsible for it , by the way we are responsible for [www.mail.yahoo.com](http://www.mail.yahoo.com) and [mail.yahoo.com](http://www.mail.yahoo.com) as well since there is no www department here in our location we manage them both".   
Actually this is exactly how DNS works : each "responsible" entity has a server (computer) that contains the relevant information in our example .com's server will store the IP of yahoo.com's server , yahoo's server will store the IP of mail.yahoo.com's server and yahoo's mail department's server will store [mail.yahoo.com's](http://www.mail.yahoo.com) IP and IP of [www.mail.yahoo.com](http://www.mail.yahoo.com) . Thus creating a chain of "searches" allowing us to find the relevant IP.  
Sharp readers may note the problem in such design , .com's server will have to deal with lots and lots or requests every millisecond since most of the sites end with .com . Thankfully a very simple mechanism fixing this issue : caching .  
Lets say that you browsed already to the [www.mail.yahoo.com](http://www.mail.yahoo.com) and now you want to refresh and see if you have any new mail , you don't really need to look for the IP again , you browser could simply remember it (and it does) , actually DNS has an option to report each step back to you or do a full recursive search by itself , but if it is reporting back and now you want to go to games.yahoo.com you don't need to start from scrach , you can simply ask for IP of games.yahoo.com from yahoo's server that your browser remembered from search of [www.mail.yahoo.com](http://www.mail.yahoo.com) . Not only your browser does caching (remembers the previous requests) the servers also cach data , so if another person will ask for yahoo.com the .com server already knows the address of yahoo's server and it does not need to look for it in his tables.  
One little thing is still not covered and it is the IP changes , new sites are created each day , sites are being closed and sites move from one location (host) to another thus changing IPs , in addition new subdomains can be created or old might be removed , all those changes require our DNS to be dynamic so lets talk about how this is covered :  
For new domains or subdomains there is no caching so the query will return the correct answer.  
For obsolete domains or subdomain , the cache of this domain will be eventually cleared since each cache uses "time to live" feature stating for how long this information is valid .  
For IP updates : eventually the DNS tables in the servers will have the correct information but if we'll access the site before the update has occurred we'll get "page not found" error .   
Actually DNS can be used for more then asking a simple "IP address" query (though it's the most common use) we can ask for aliases of some site , ask for the mail exchange server etc ... but it's out of the scope of this exercise.  
  
I hope you enjoyed reading (and maybe even learned something) , thanks for your time !  
(I'm not native English speaker so I'd appreciate if you could ignore grammar mistakes :) )