1. Search: "firmware" at Google

<https://www.google.com/search?sourceid=chrome-psyapi2&ion=1&espv=2&ie=UTF-8&q=firmware>

. . .

. . .

2. Search results are returned within 0.36 seconds.

At the search engine side:

When we type a keyword to search in google, it is not actually searching all of the web in run time to provide us with the result, but instead it is only searching its **indices to the web**, accumulated by google search engine’s web crawler/spider.

**Web crawler/Spider**: A web crawler is an application that starts off by fetching a small subset of popular web pages, then follow the links in those pages to fetch them, and follow the links in this 2nd level of pages and so on. This process is iterated till as big a chunk of the internet as possible.

**Indices & caching**: The webcrawler assigns keywords of phrases to each of the page it has thus fetched. These indices and cached copies of the web pages are stored in thousands of google servers.

When we search the term “firmware” on internet, google search engine is actually visiting all those indices to find the most relevant pages ( depending on **how many times the term appears** in the page, where the **links** in those pages are pointing, checking whether those pages are **spamming** etc.) and also looking at the “**page\_rank**” of the page. Page rank is number that is determined based on number of outside links to particular page, and the relevance/importance of pages that carry those links. Once it comes up with a subset of those indices, the subset is populated to the user in the form of search results.

Very basic summary of what happens from the Networking point of view:

1. Web browser looks for I/P Addr of google.com in the following caches one after the other till it gets the IP.
   1. Browser’s DNS cache, if it fails then
   2. Kernel’s DNS cache, if it fails, then
   3. Sending DNS req to router/DNS Server
2. Once IP Addr is obtained it initiates a TCP connection.
3. When the IP layer has to encapsulate to send this to the peer, it first looks in arp cache to find the EUI48 corres. to IP address.

* If that fails it send arp request
* Router will respond as a proxy arp

1. The TCP Connect is then sent out encapsulated in MACHdr,IP Hdr.
2. The router performs NAT and sends it to the ISP server which then routes forward the request.
3. Once the TCP connection is established Browser sends HTTP request to retrieve the web page.
4. Browser gets HTTP Response from the server and displays the page content in the browser.