# *Proxy Server*

# *Design Document*


***Schematic View of the Network***

# *Procedure:-*

1. First the client will send a request to the proxy server.
2. Then the proxy server forwards the request that was received from the client to the host server in the name of itself in place of the client.
3. Then the host server replies back and serves the request of the proxy server by sending the appropriate page to the proxy server.
4. Then the proxy server after receiving the response from the host server replies back to the client who requested it by sending it the page it received from the server.
5. In this process proxy server takes advantage of the fact that it can store information and send them to clients when they request the same page again.
6. That is when a request is made for a page the cache in the proxy server first looks up in the table it has of all the pages it has.
7. In case the request is found to be in the cache the proxy server sends a last modified request to the host server.
8. Then the response date and time is compared with that in the cache of the proxy server.
9. Using this cache it has stored the proxy server finds whether the page is brought again from the server or not.

10. In case the page was not modified from the time it was brought to the server the proxy server sends the page stored in the cache to the client requesting the page.
11. In this way the load on the host server is reduced as well as the band width is saved.
12. The algorithm that can be used to serve the cache is first in first out algorithm.
13. This algorithm helps in the way that mostly the urls that were accessed much earlier in the stack would be less used.
14. Thus using the First in first out algorithm we can save a lot of querying into the server for the page and page hits can be minimal.
15. It could be slightly better in case we use last recently used algorithm but not heavily significant when we have a cache of size 100.
16. A log of all the queries is maintained in the log file.
17. It is helpful in keeping track of actions that have taken place recently in the proxy server.
18. The get request that is sent by the client is forwarded by the proxy server to the host server.
19. Thus proxy server acts as an intermediate router between the client and the server.
20. Proxy server helps in making the transactions through the internet from a particular location fast.

21. When a conditional get request from the client is issued to a proxy server even though the element is in the cache it cannot be served directly to the client.
22. In the case of a client requesting a conditional get request, the proxy server receives the url from the client and then looks up for the url in the hash table that stores the urls. After the lookup in case the url is not found in the cache it will be no different from the non – conditional get request. But if the url that the client sent to the proxy server is already existing in the has table then the proxy server will send a conditional get request to the host server and gets the time stamp.
23. In case the timestamp is same in both the cases then the proxy server does nothing but serve the page it has in the cache.
24. But in case it is different, then the proxy server requests for the page from the host server and then returns the new page to the server that has the modified information and also updates the cache with the current copy of the url.