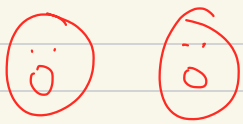
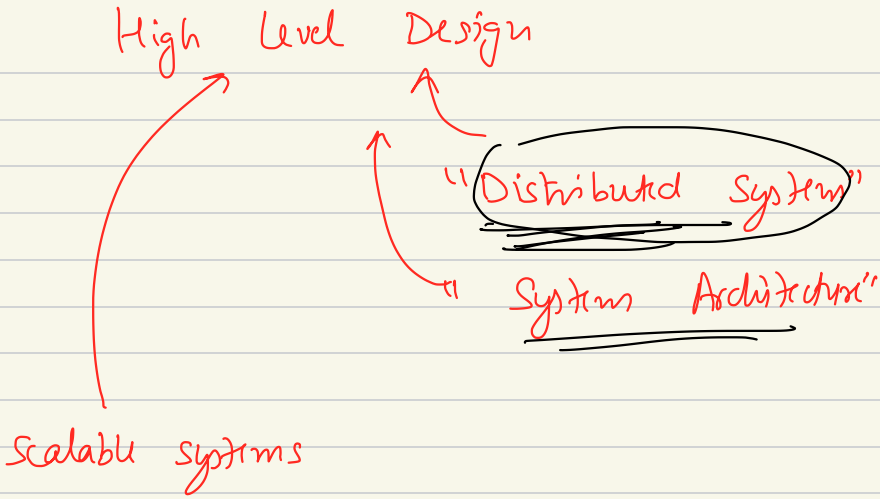
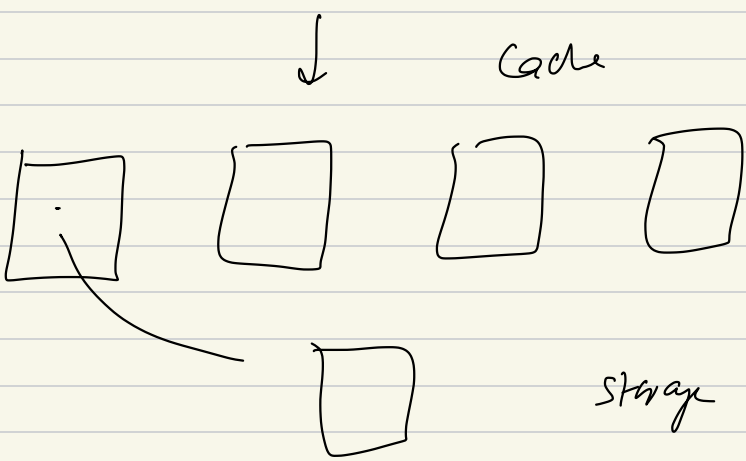
High Level Design

1. Respectful of the class's time.
   → use chat section only in case of super urgency.
2. Questions Tab ≡ Doubts
3. HLD — vast 
   Focus on foundations
4. Yes No 



Designing / Creating systems at scale.

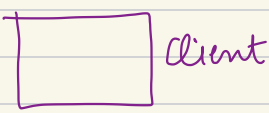


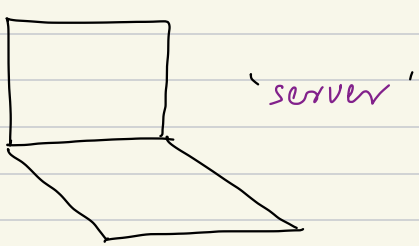
delicious

2000s

Bookmarks

- add Bookmarks
- view Bookmarks





IP addresses.

public

private

Static dynamic

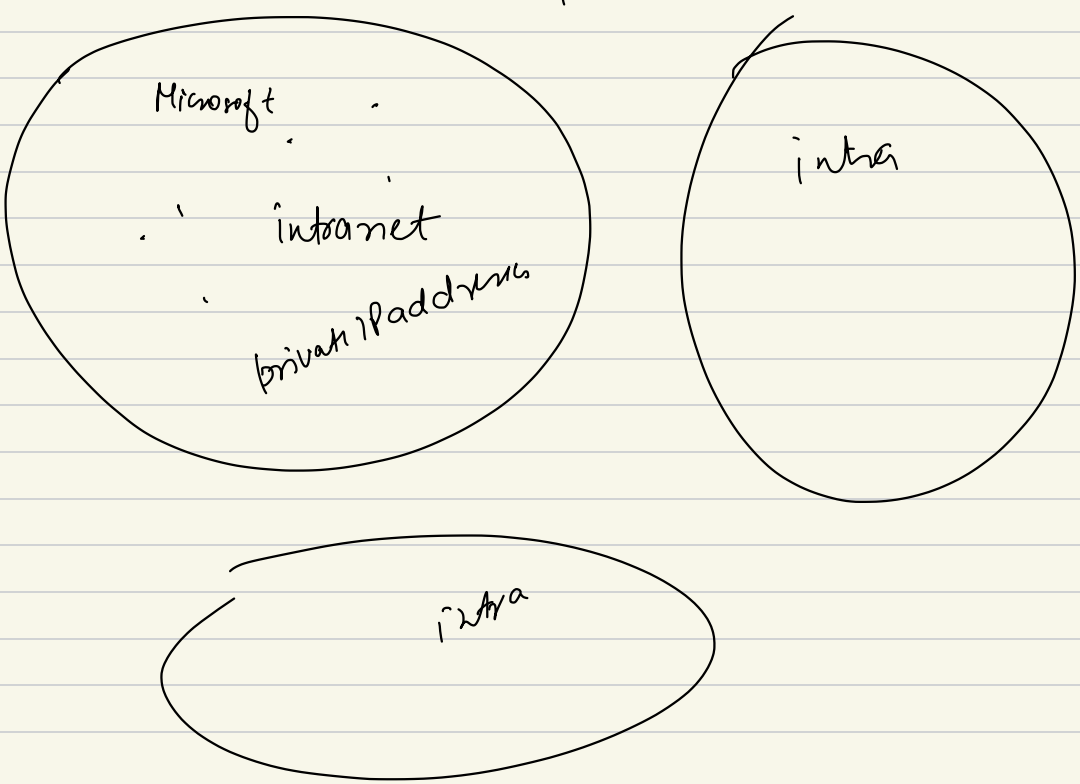
MAC add vs IP addr

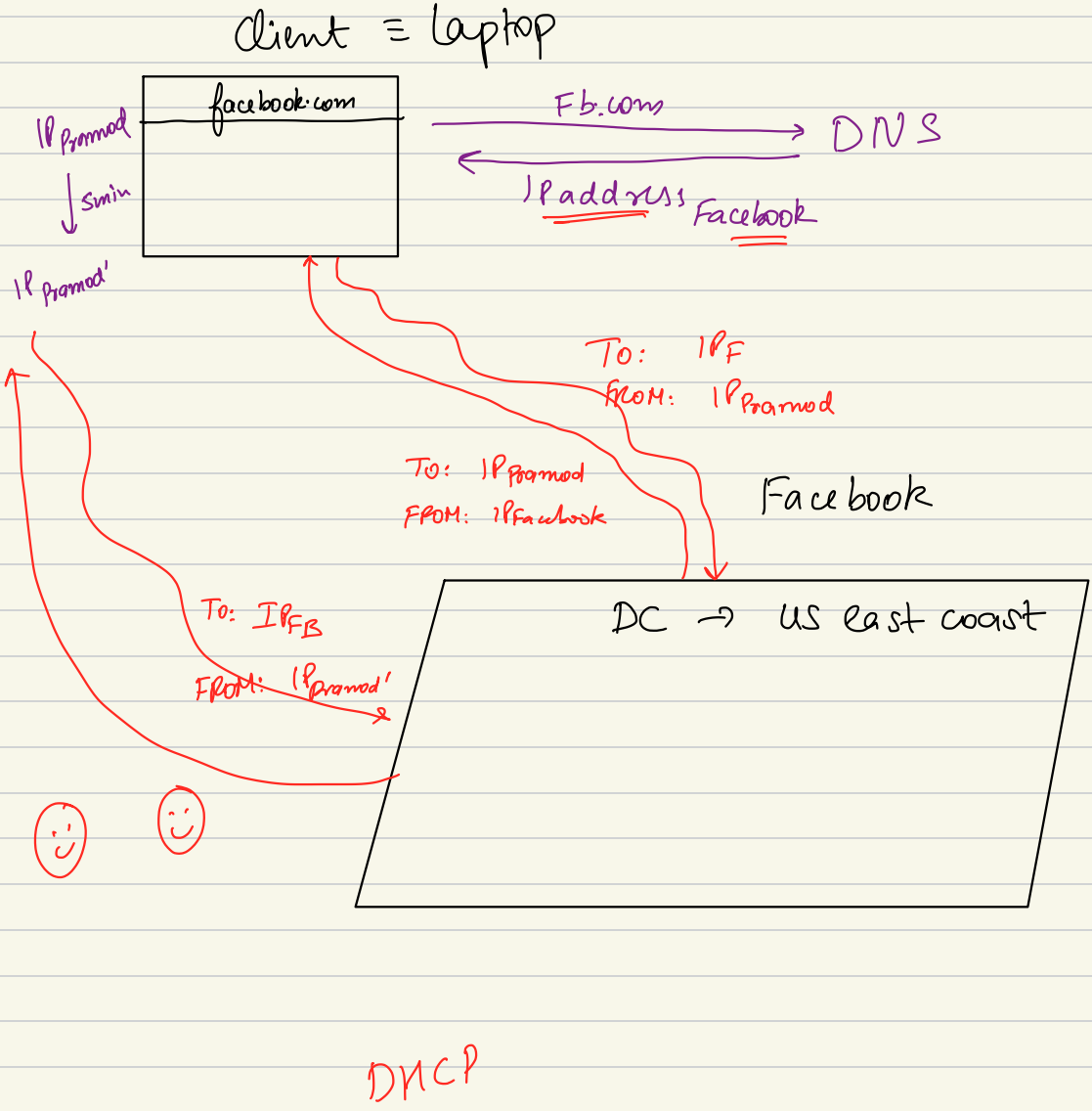
biometrics house add

---

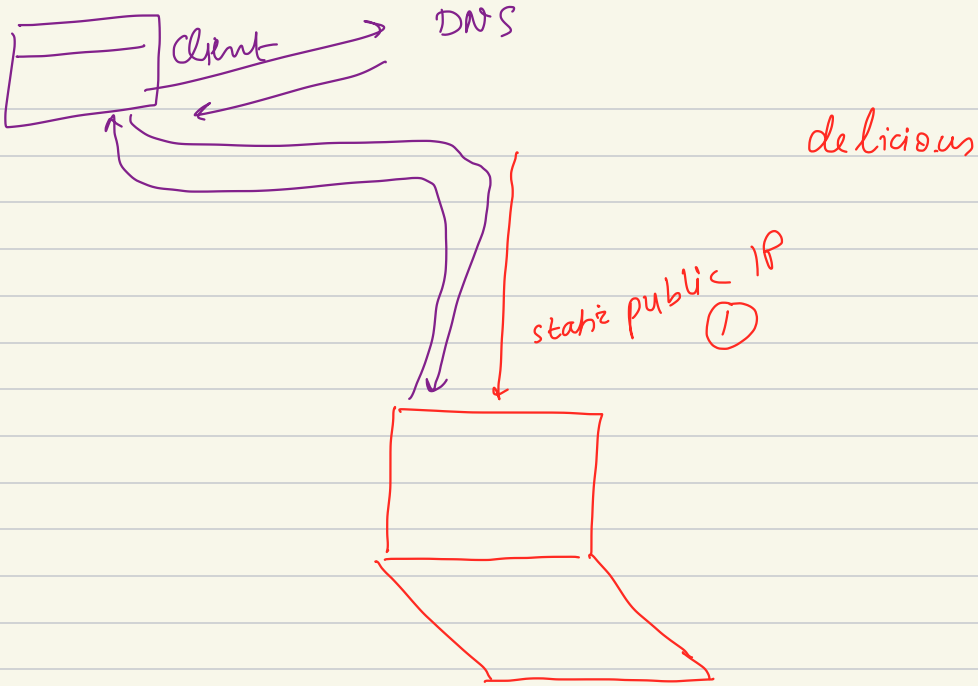
internet → (network of networks)

public IP addrs




Client = laptop
facebook.com
IP Pramod
5min
IP Pramod'
Fb.com
DNS
IP address Facebook
To: IPF
FROM: IP Pramod
To: IP Pramod
FROM: IP Facebook
Facebook
DC → US east coast
To: IPFB
FROM: IP Pramod'

DHCP



(2) DNS Registration

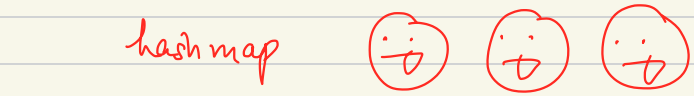
(Domain Name Server)

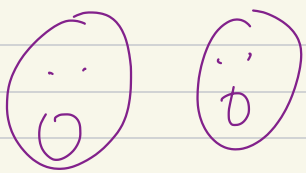
domain	IP
Facebook.com	→ IP FB

DNS

Domain Name System

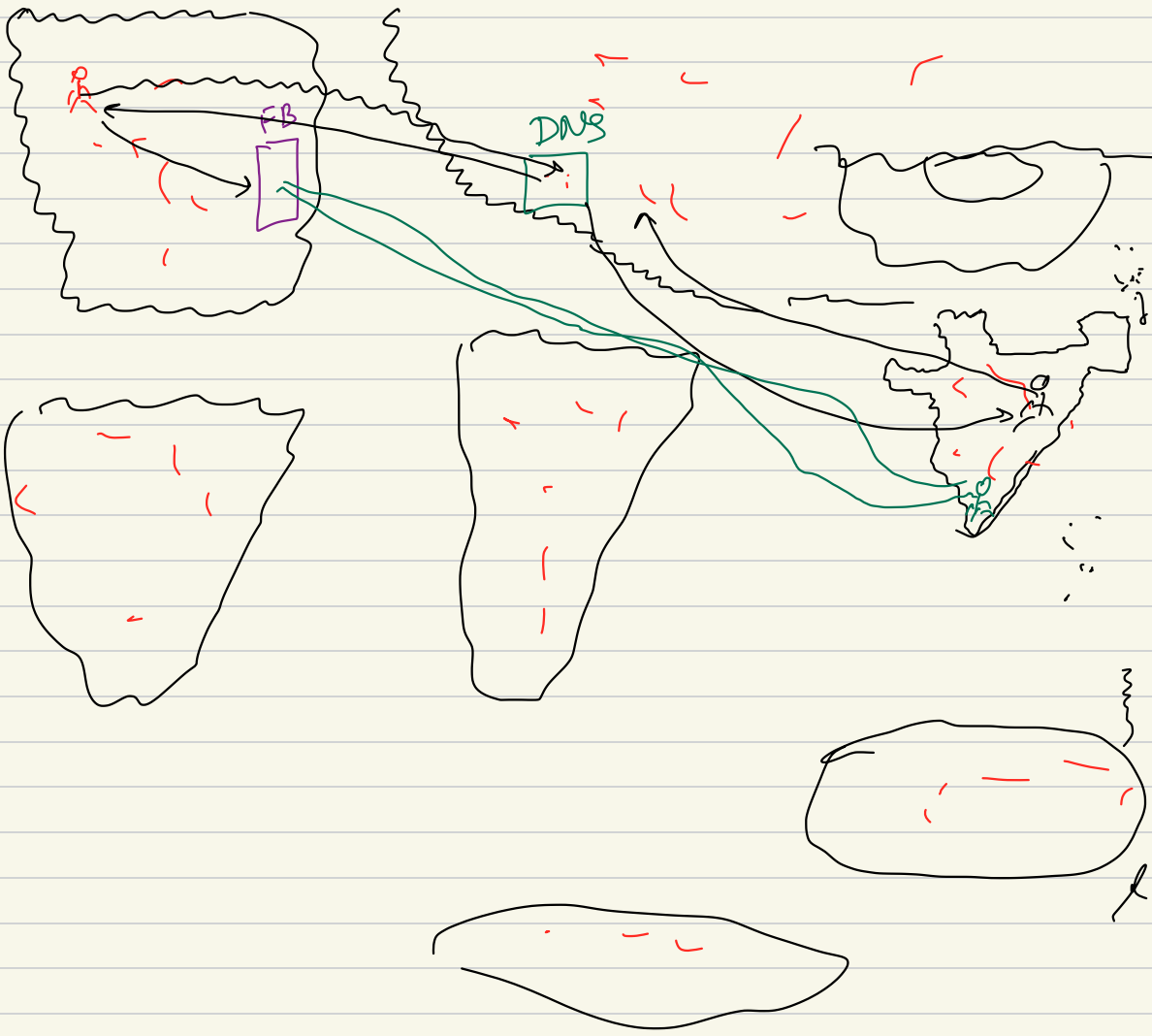
hashmap





System architecture

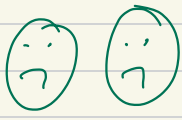
giant


FB
DNS

DNS

↳ in reality needs to handle requests at the internet scale

Problems:

1. ✓ scale 
2. ✓ latency 

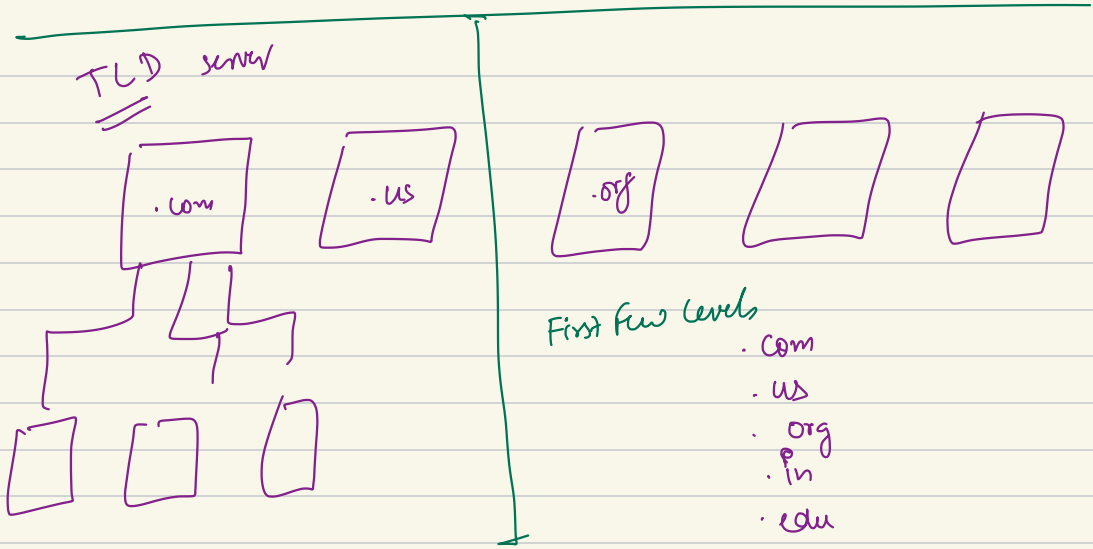
latency ∝ distance

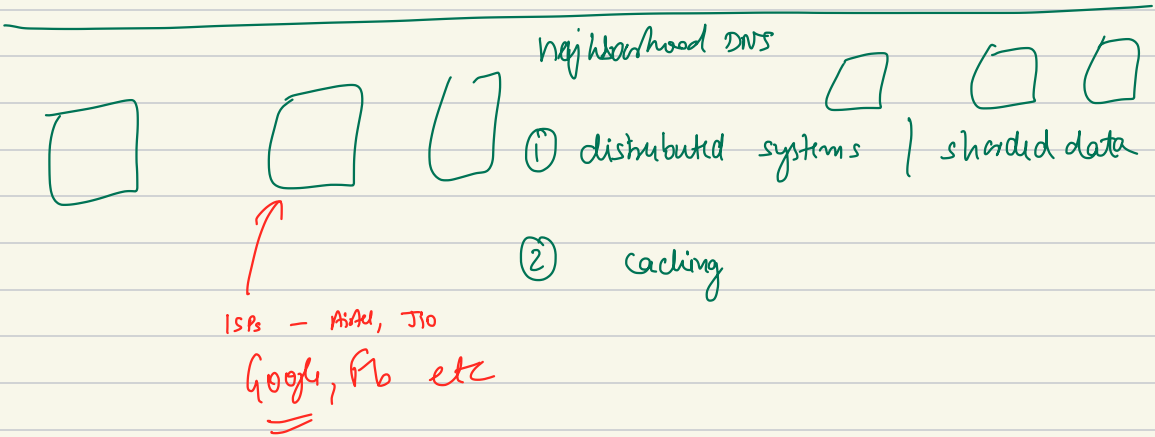
11 pm ——— 11:50 pm

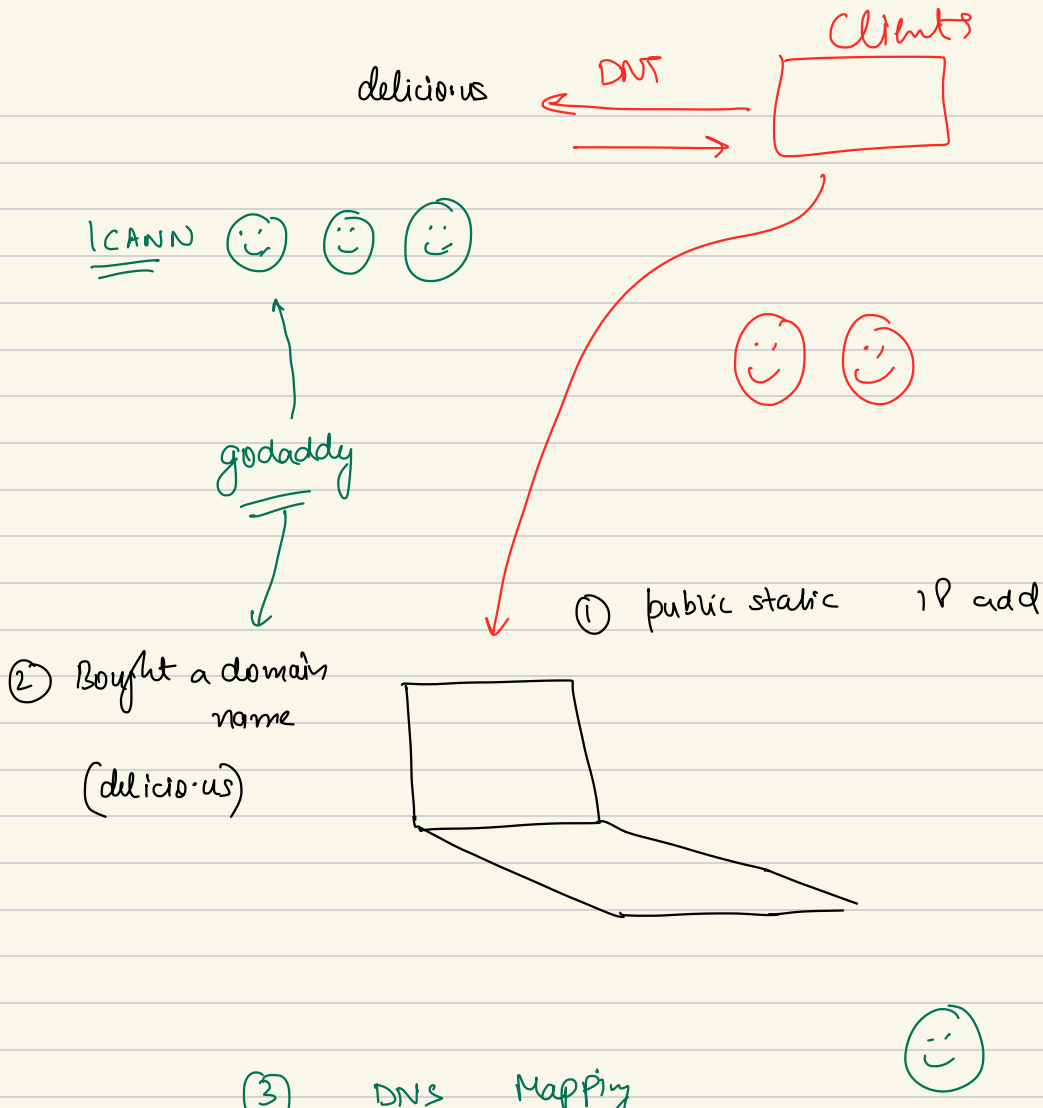
Doubts

Break
11:08 pm

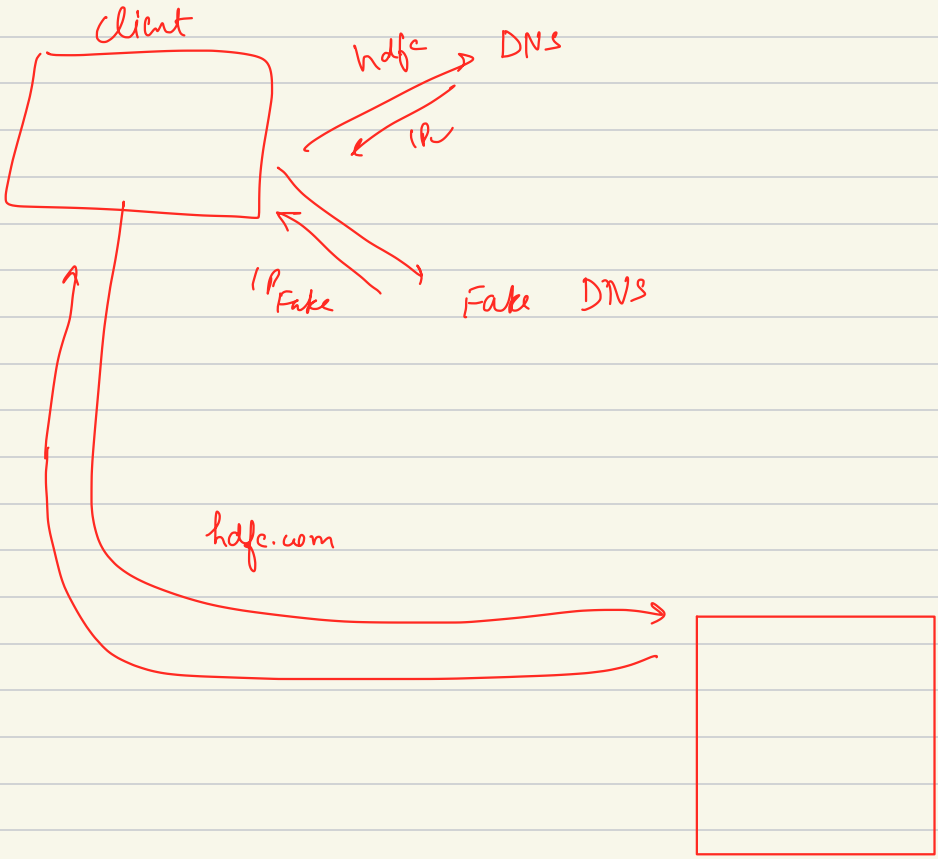
DNS → distributed system


TLD server
.com
.us
.org
First few levels
. com
. us
. org
. in
. edu


neighborhood DNS
① distributed systems | sharded data
② caching
ISPs – Airtel, Jio
Google, FB etc


Clients
delicious
DNT
ICANN
godaddy
① public static IP add
② Boyght a domain name
(delicio·us)
③ DNS Mapping
domain ↔ IP

DNS spoofing



delicious became popular.  

So many bottleneck can arise

1. CPU bottleneck
2. hard disk bottleneck
3. RAM bottleneck
4. network card bottleneck
5. SPOF → single point of failure

HORIZONTAL SCALING

OR

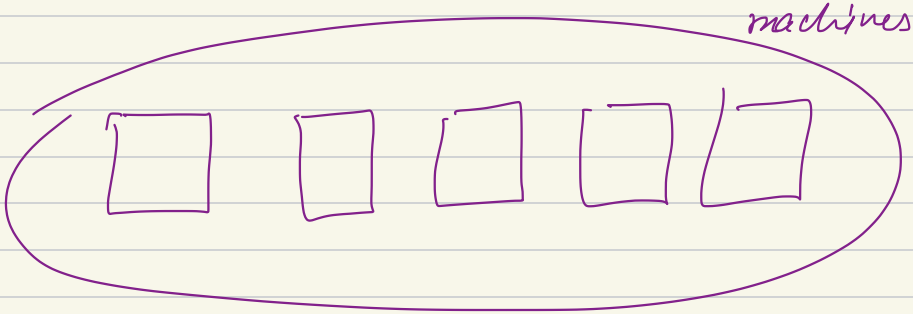
VERTICAL SCALING

I VERTICAL SCALING

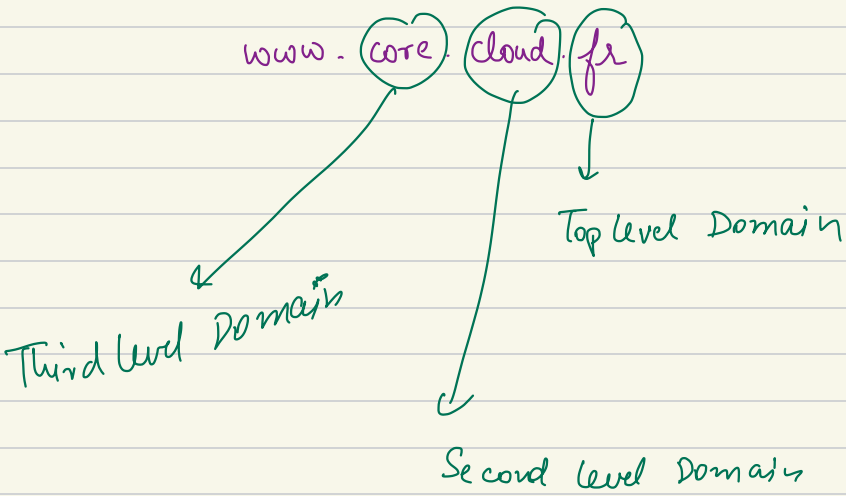
machine ≡ node ≡ server

II HORIZONTAL SCALING

→ add more small/cheap machines



O – A session


www.
core
cloud
fr
Top level Domain
Third level Domain
Second level Domain