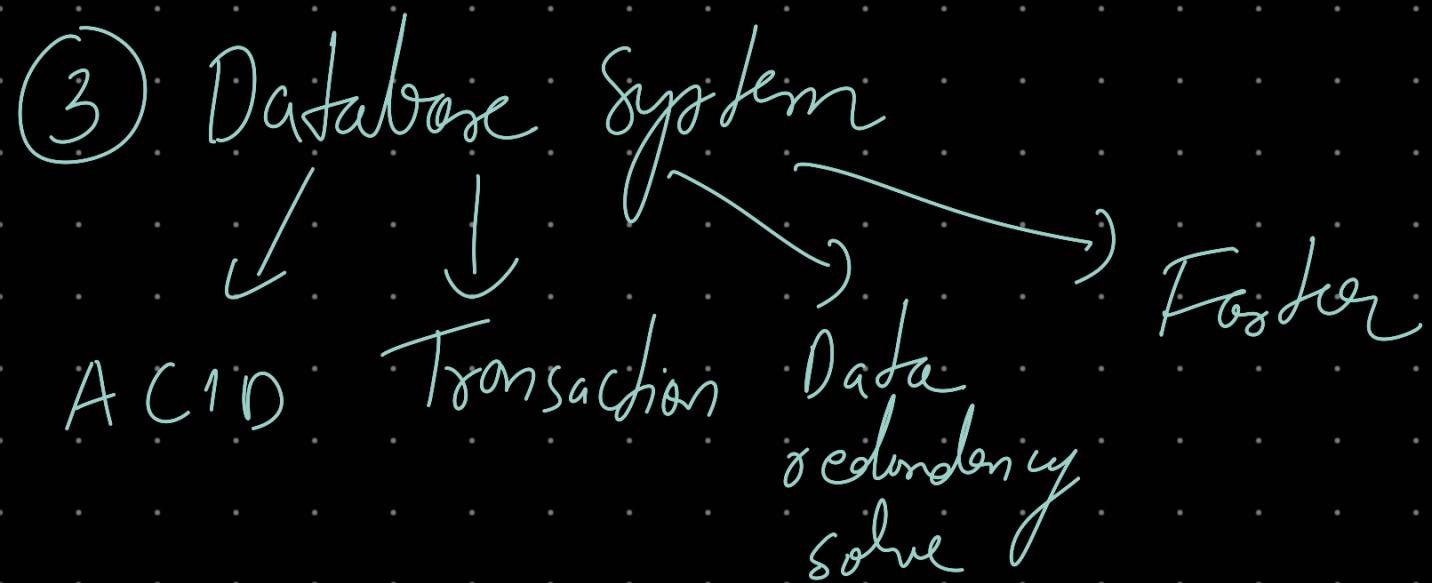
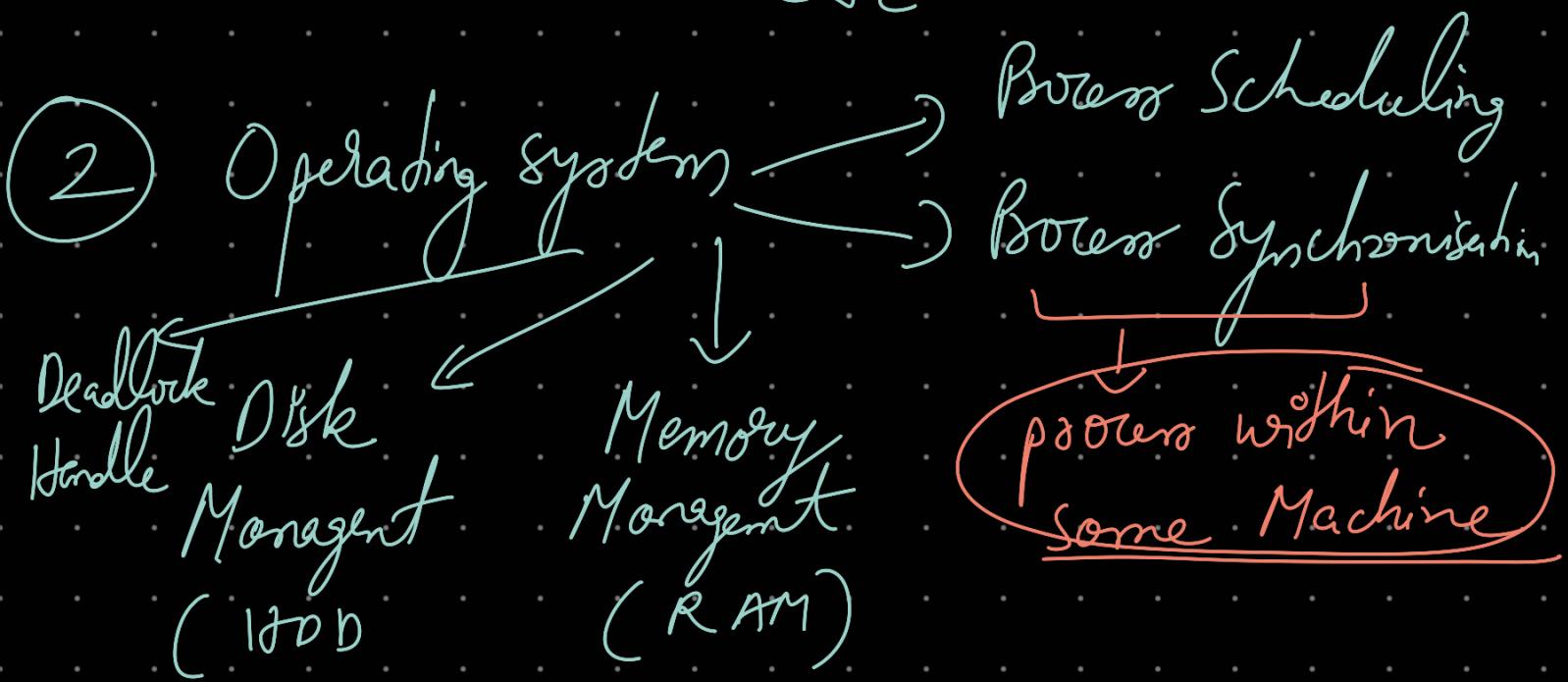


Basis of Distributed Systems

① Hardware → CPU, RAM, HDD, NIC card etc



④ Hardware + OS \Rightarrow One Machine



How to transfer data?

Hardware + OS \Rightarrow Second Machine

\rightarrow Networking stack + protocols

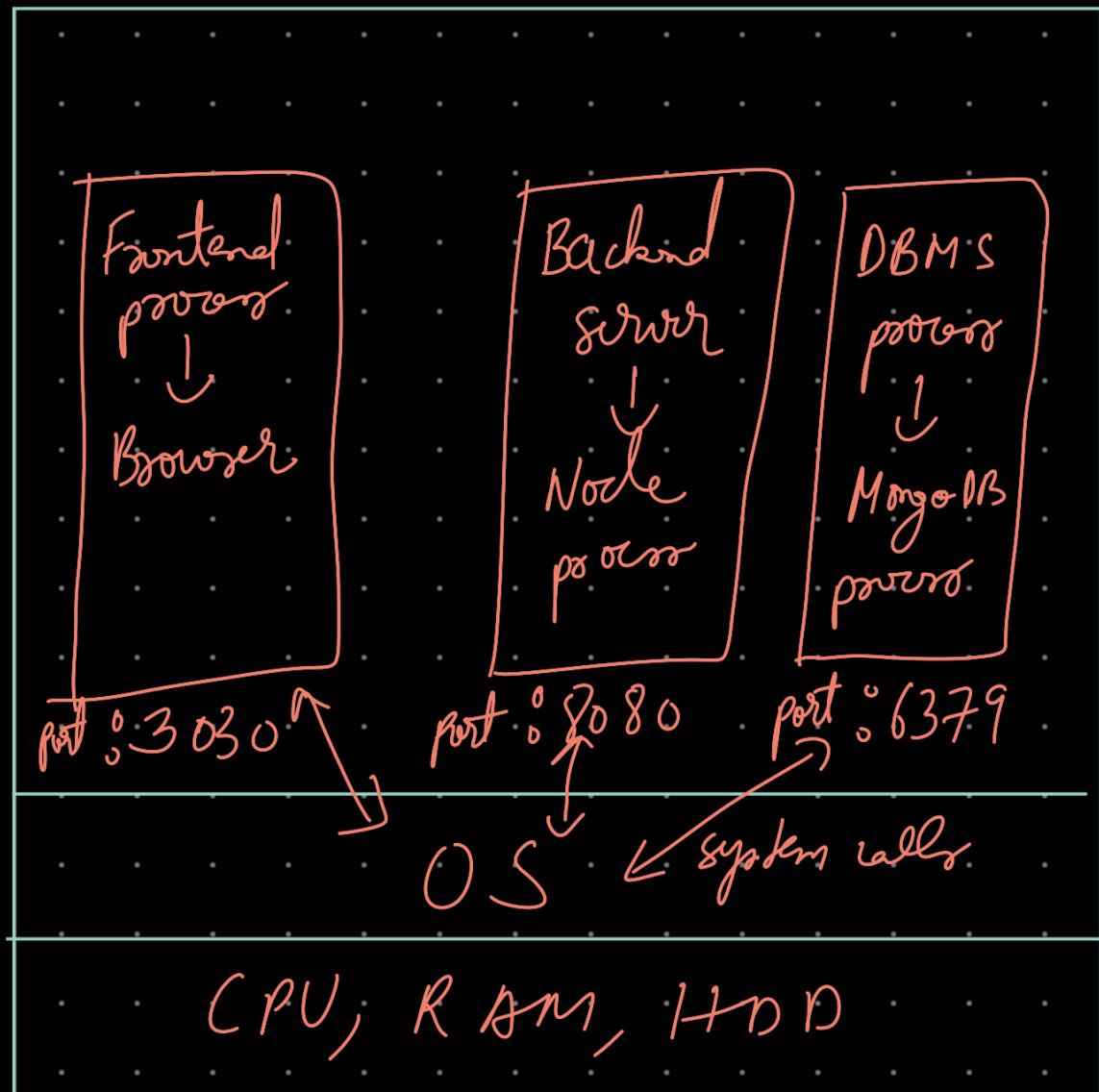
(TCP/IP stack)

Consider how we develop application

Lorally



Bare Minimum Architecture

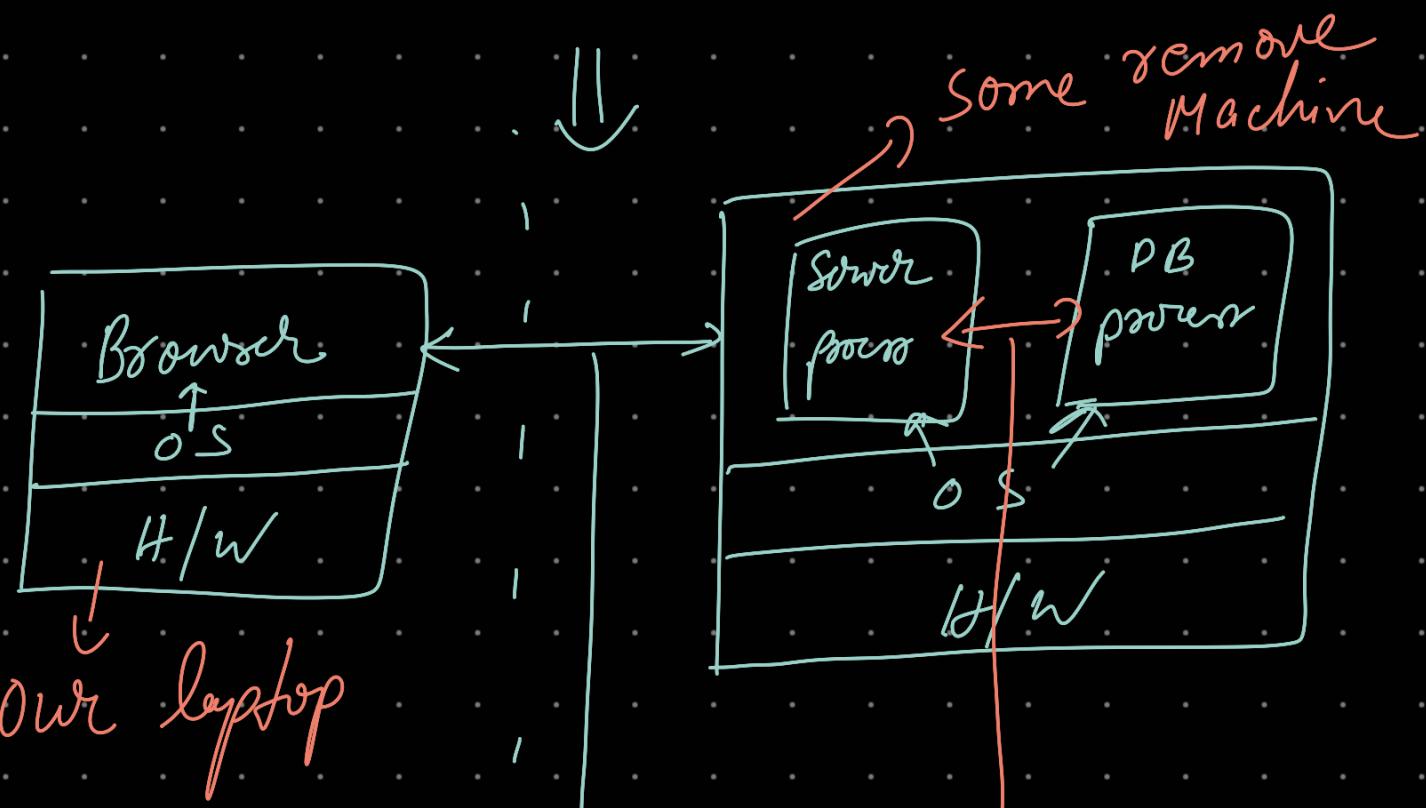


* All three are in same Machine
 Only one request is going i.e we
 ourselves → no concurrent request

↓

Here everything works
 fine

We deploy it to be available on Internet



Networking

These are communicating using

IPC or unix domain socket.

Now this Remote Machine is accessible on Internet

20 concurrent user



2000 users



2 lac users



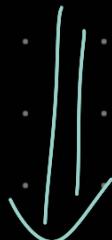
2 M users and so on



Here our server will start
facing lots of problems



RAM ↑ DB ↑ CPU ↑
usage usage usage



First step is to best optimise
your code with indices, optimise
query, tune DB



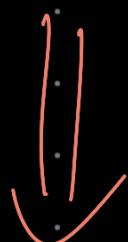
Do vertical scaling (Increase RAM
(CPU, HDD))

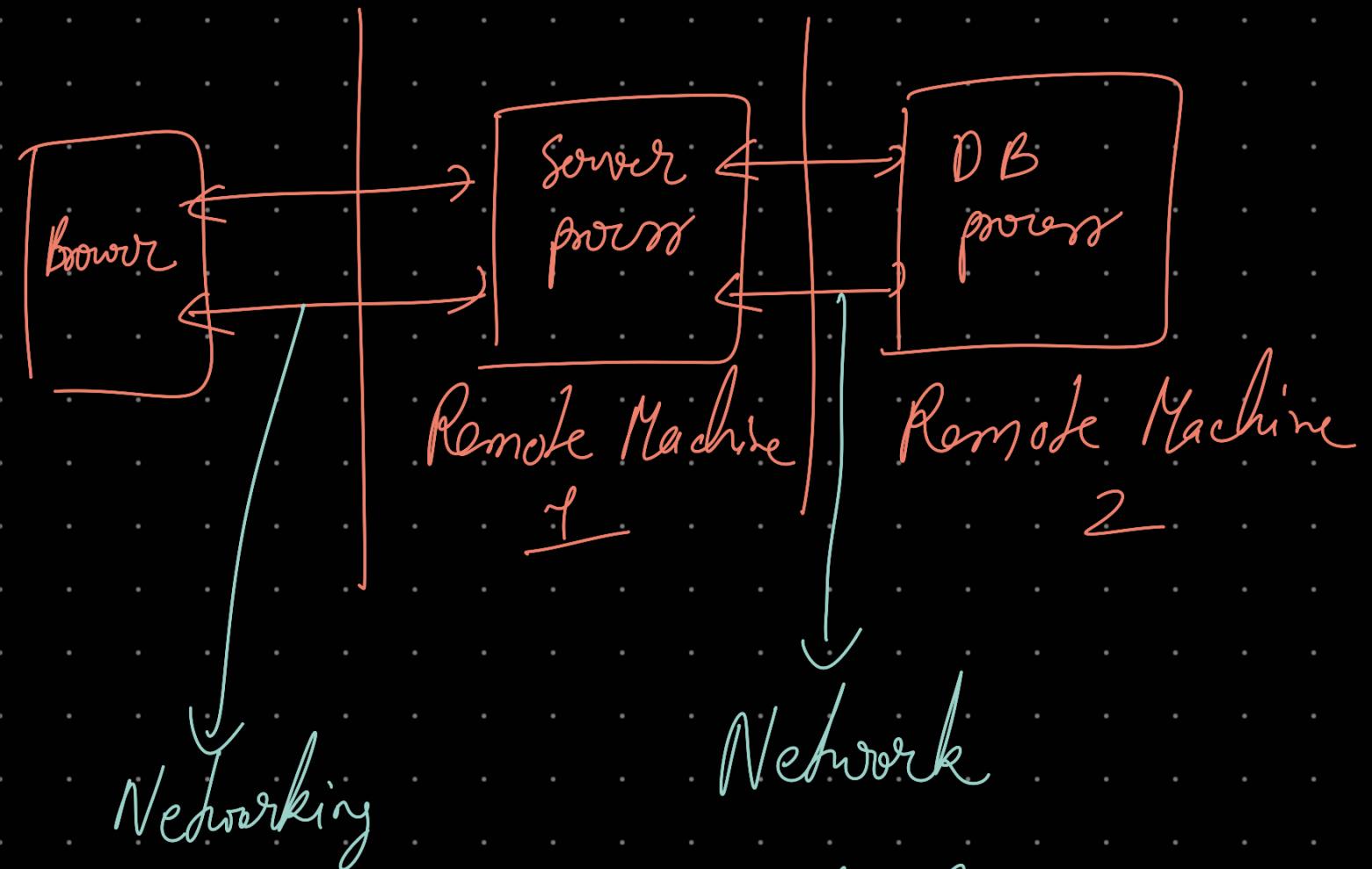


Once everything is done



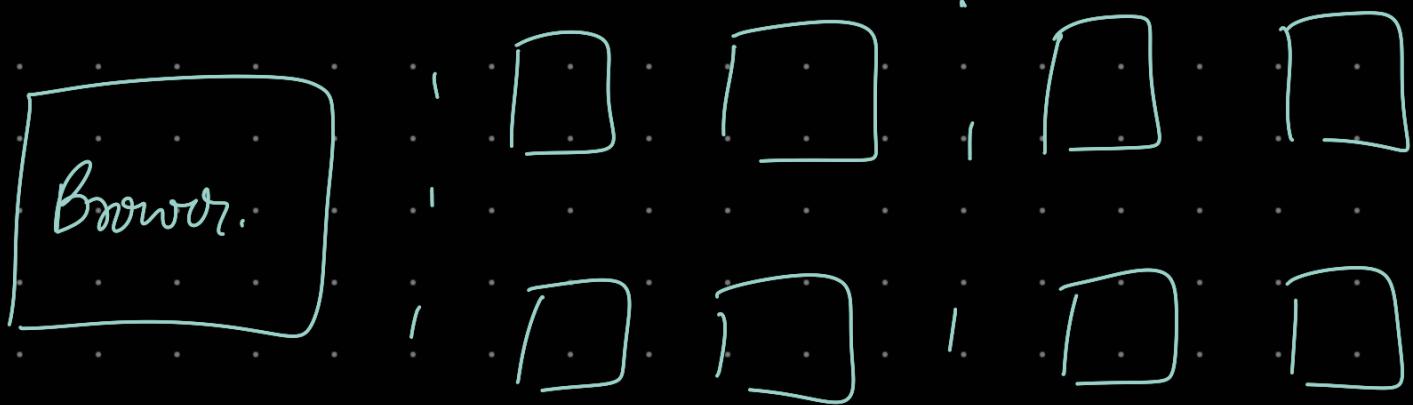
Decide System





Now, some more challenges will come, latency is main challenge

Further scaling



Multiple server
Machine

Multiple Machine
running DBMS

More complex and challenges

* According to scale of our application, we design system