

• lec 2 Distributed system :-

• Distributed system challenges :-

- Parallelism (what about cpu, memory, I/O)
- Fault tolerance
 - Availability.
 - Recovery.
- physical
- Security (Data protection / Isolation)

Hint

the goal of Distributed system (DS) provides

‘infrastructure for applications.’

• might be one of the different Applications

following :-

1. Storage

2. Communication

3. Computation

عشان ميتعملوا
الويس مع بعض
محتاجه

well define interface

لا تاني جازي يكون موجود عنده
مختلف . وبالتالي اننا هنا محتاج الحققة

جزء من openness

we need to Abstraction

it means

“hide the distributed nature of sys.”

RPC
thread
Concurrency
Control.

MapReduce
is a good example
of this element.

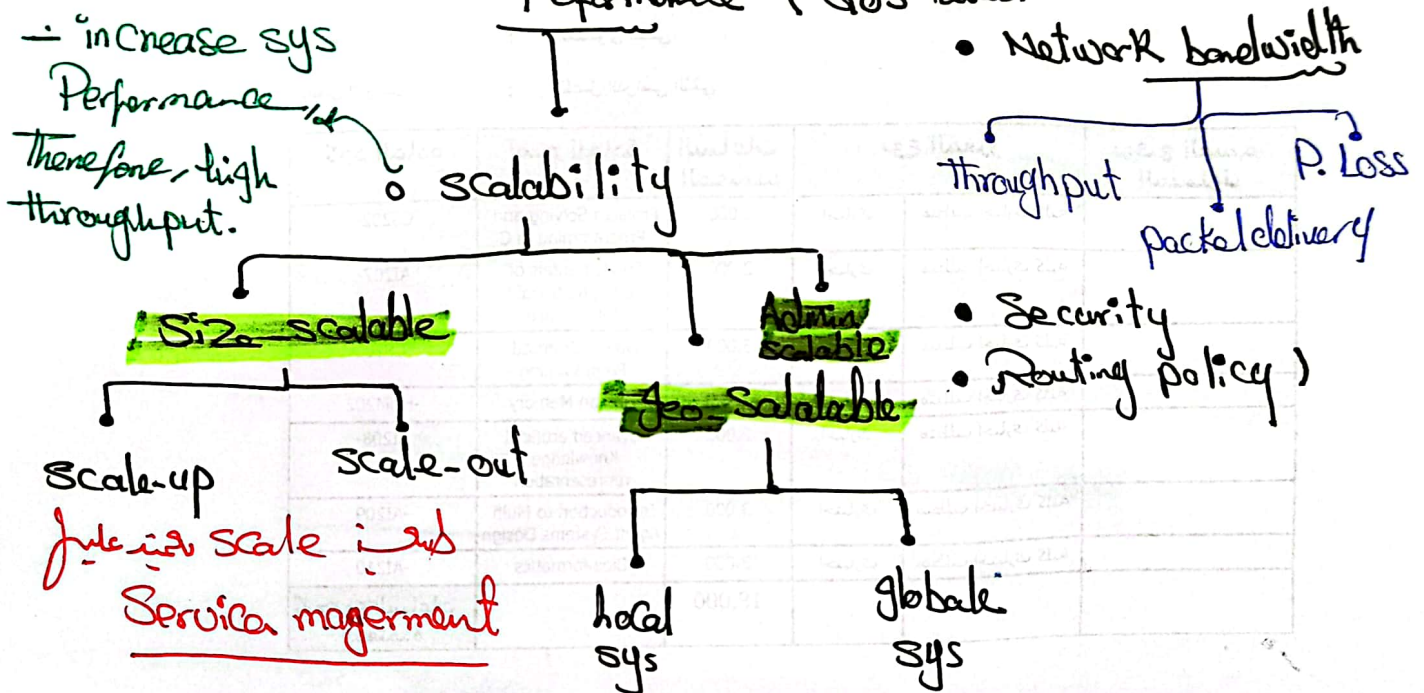
Partial failure ~ ~ ~ ~ ~

◦ when some items are **stop working** other items are **continue working**.

◦ in general the **partial failures** occur by the network infrastructure / Connection unreliable.

◦ Three main factors to overcome DS's challenges :

① - Performance (QoS level)

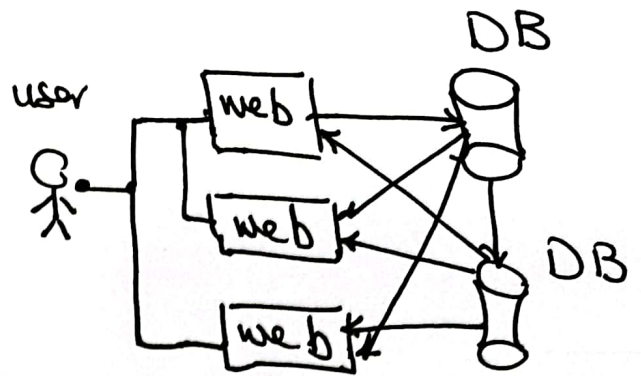


Scalability \Rightarrow High sys performance \Rightarrow High throughput

② Fault tolerance.

- are significant requirement of DS.
- Availability. (partial failure)
- system recovery. (Repairing for crashing Parts without any loss of correctness)

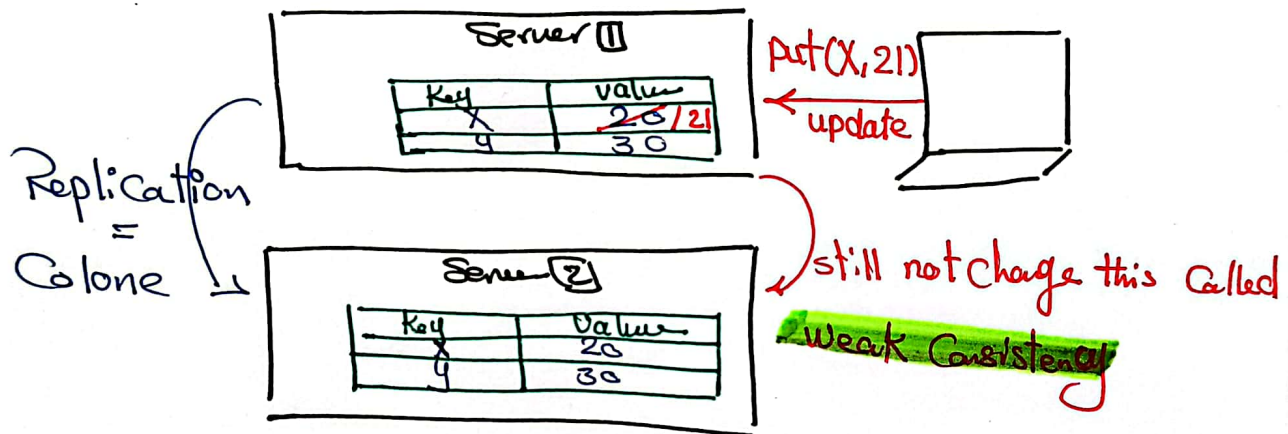
◦ example of scalability:



[3] Consistency — Strong Consistency
Weak Consistency (useful)

EX:

— Put (Key, value)
— Get (Key) } way to Read/write the app among servers.



— in a real case, ~~support~~ DS supports weak consistency more than strong consistency.

hint

strong consistency in DS :

- very expensive
- sys. needs a lot of chitchat to read one value. (network bottleneck)