

NON FUNCTIONAL : ARSMF

① AVAILABILITY: % of time , service is accessible

100% avail = responds all the time

$$A(\text{in\%}) = \frac{(\text{Total time} - \text{Amount of downtime})}{\text{Total time}} * 100$$

- Availability measurement varies across providers.
- Some start from the service's launch.
- Others begin from a specific client's usage.
- Reported availability may exclude:
 - Downtime affecting only some clients.
 - Planned maintenance periods.
 - Downtime caused by cyberattacks.
- Important to analyze how each provider calculates availability metrics.

② RELIABILITY (R)

- Probability that service will perform functions for a specified time
- R measures how service performs under varying operating conditions

MTBF: Mean time between Failures

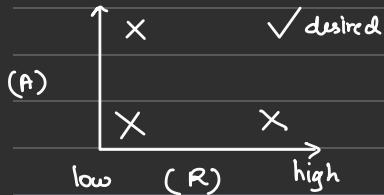
MTTR . Mean time to Repair

$$MTBF = \frac{\text{Total time elapsed} - \text{sum of down time}}{\text{total num of failures}}$$

$$MTTR = \frac{\text{Total maintenance time}}{\text{total num of repairs}}$$

Availability Percentages versus Service Downtime			
Availability %	Downtime per Year	Downtime per Month	Downtime per Week
90% (1 nine)	36.5 days	72 hours	16.8 hours
99% (2 nines)	3.65 days	7.20 hours	1.68 hours
99.5% (2 nines)	1.83 days	3.60 hours	50.4 minutes
99.9% (3 nines)	8.76 hours	43.8 minutes	10.1 minutes
99.99% (4 nines)	52.56 minutes	4.32 minutes	1.01 minutes
99.999% (5 nines)	5.26 minutes	25.9 seconds	6.05 seconds
99.9999% (6 nines)	31.5 seconds	2.59 seconds	0.605 seconds
99.99999% (7 nines)	3.15 seconds	0.259 seconds	0.0605 seconds

- SOL : service level obj
- Health of device - avail
~ reliable
- Measurement of A: timeloss
- measure of R: freq & impact of failure
- A & R are related
- A is a function of R
- \Rightarrow R is independent
A depends on R



- ③ SCALABILITY: — vertical
— horizontal

- ✓ Increase workload same performance
- no. of reqs, Data storage
- ★ SIZE: new users to share DS
- ★ ADMINISTRATIVE: more orgs share DS
- ★ GEOGRAPHIC: new region

- ★ VERTICAL: scale up! more CPU, Ram.
expand current hardware & software
\$\$\$ → high
- ★ HORIZONTAL: scale-out! no. of machines
Many nodes work as ONE.

(4) MAINTAINABILITY :

After building system:

- find, fix bugs
- Add features
- Updated platform

* OPERABILITY: smooth operation, no faults

* LUCIDITY: simple code. Understand & maintain

* MODIFIABILITY: new features

Measuring M:

- M is the probability that service will restore to its functions within a Specified time of fault occurrence.
- regains normal operating conditions.

$$MTTR = \frac{\text{Total maintenance time}}{\text{Total number of repairs}}$$

(\downarrow mean time to repair

(5) FAULT TOLERANCE

Replication

check pointing.

- Large scale systems - 100s of DB, servers billions of user requests
- Data safety & no recalculation of CPU intensive tasks
- FT = system's ability to execute persistently even when 1 component fails.
- component \rightarrow hw or sw

* REPLICATION: swap failed nodes w healthy ones

multiple copies : update regularly (challenge)
reduce availability.

Asynch updates: eventual consistency. } avail vs consistency

CHECKPOINT • saves system state in stable storage.

• previously functioning check point roll back.

• consider 3 processes: i, j & k

2 msgs are exchanged.

one snapshot / checkpoint: C_{1i}, C_{1j}, C_{1k}