

Non-functional requirements - Scalability

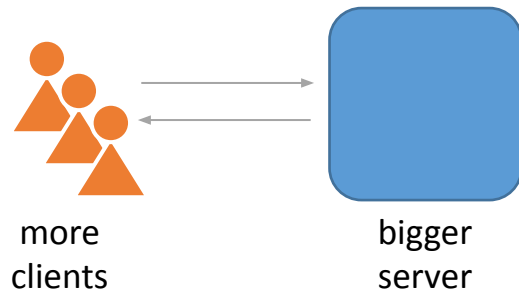
scalability

- requests per second
- volume of incoming or outgoing data
- number of concurrent connections
- other...

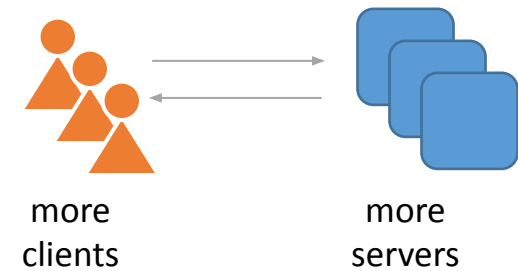
the property of a system to handle a growing load

Non-functional requirements - Scalability

vertical scaling (scaling up)

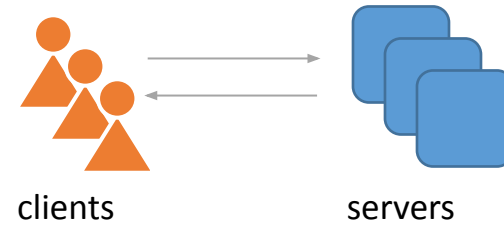


horizontal scaling (scaling out)



shared-nothing architecture

Non-functional requirements - Scalability



horizontal scaling comes along with challenges

service discovery

load balancing

request routing

maintenance

Non-functional requirements - Scalability

I still think that every system must run on multiple machines these days. It allows to have redundant machines in the system. Thus, increases availability.



software
engineer

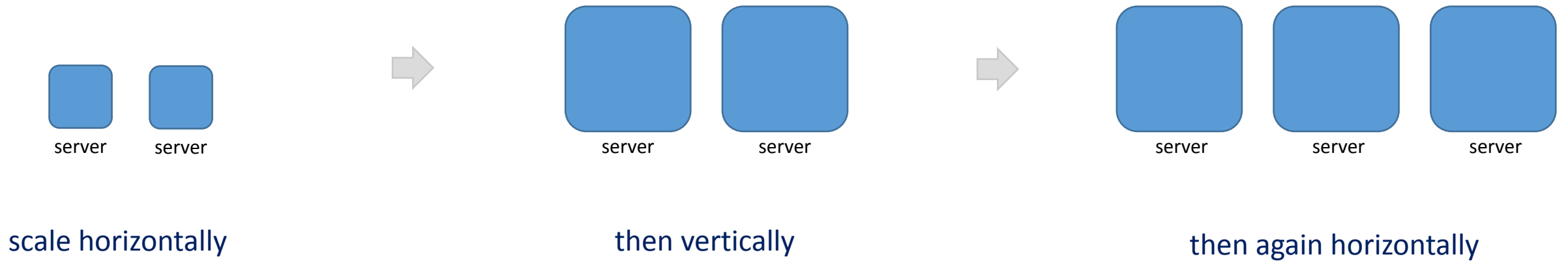
It is possible to achieve high availability in a single machine (almost 😊) system. Relational databases and load balancers often use active/passive setup for this.



software
engineer

Non-functional requirements - Scalability

horizontal and vertical scaling can be used in conjunction

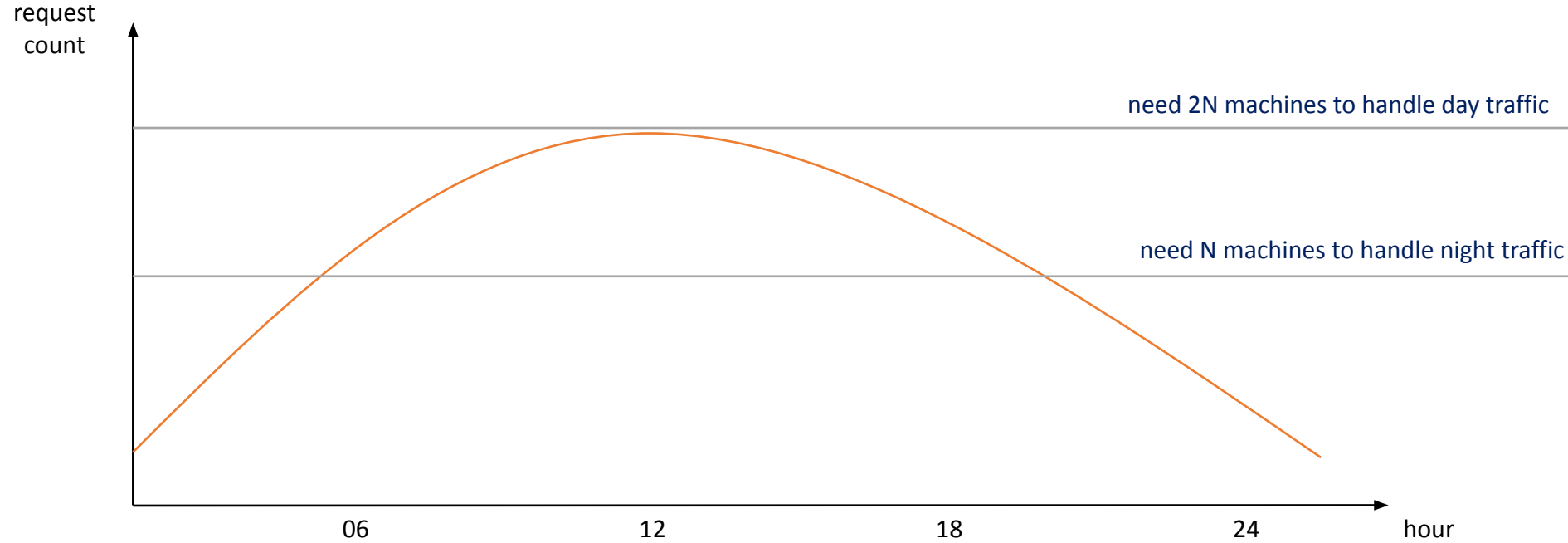


Non-functional requirements - Scalability

elasticity \neq scalability

the ability of a system to acquire resources as it needs them,
and release resources when it no longer needs them

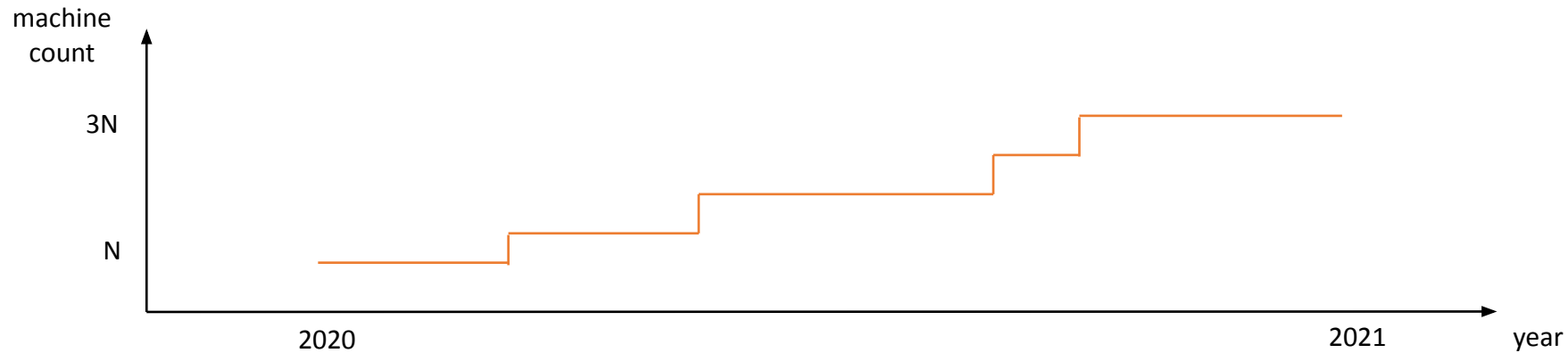
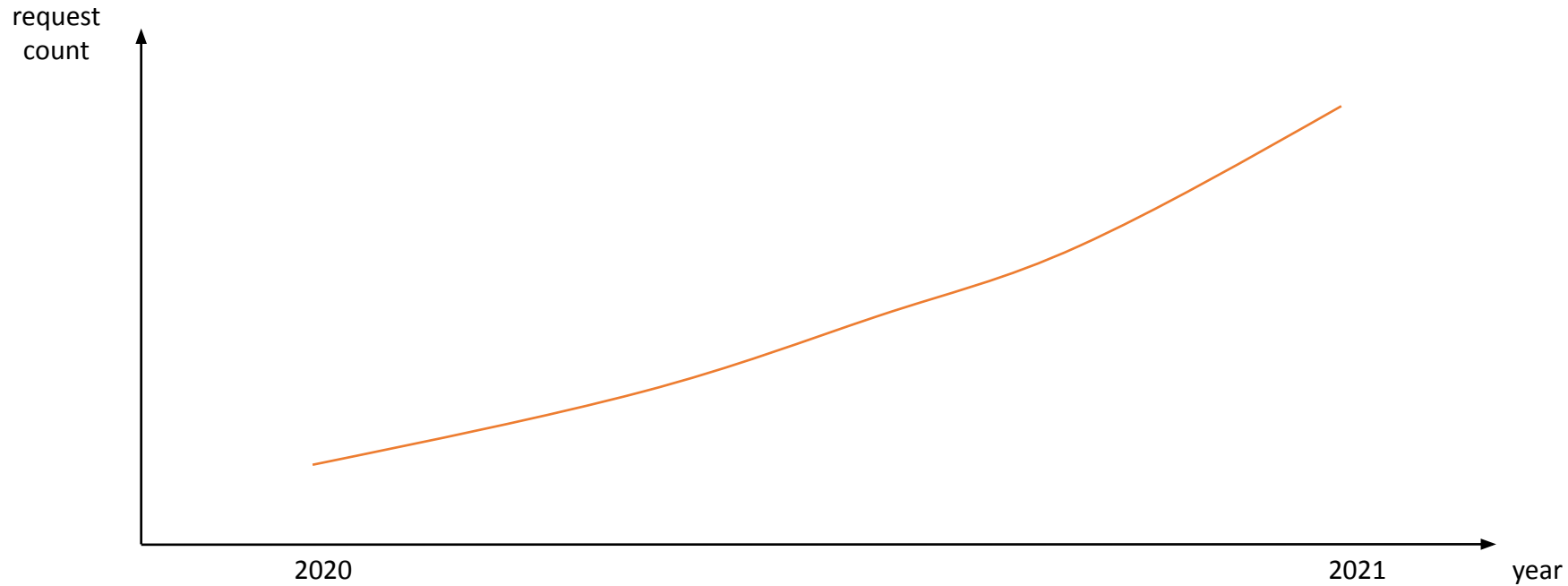
Non-functional requirements - Scalability



elasticity

we **grow** or **shrink** infrastructure **resources**
over **short time intervals** to adapt to workload
changes

Non-functional requirements - Scalability



scalability

we **provision resources** in incremental manner over **longer time intervals**

Non-functional requirements - Scalability

scalability

is about long-term, strategic needs

is required for

elasticity

is about short-term, tactical needs