Fault tolerance

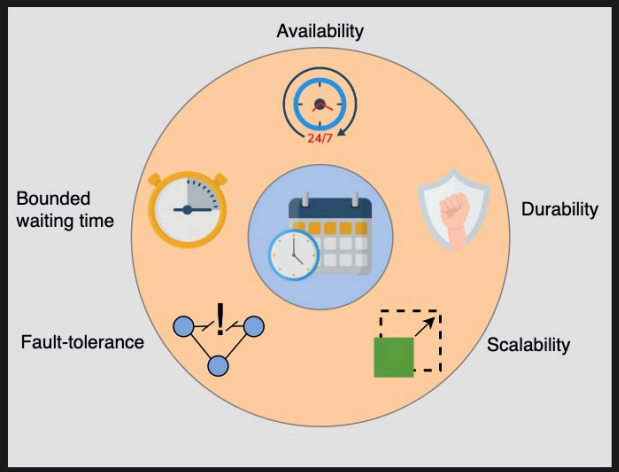
A task is not removed from the queue the first time it is sent for execution. If the execution fails, we retry for the maximum number of allowed attempts. If the task contains an infinite loop, we kill the task after some specified time and notify the user.

Bounded waiting time#

We don't let the users wait for an infinite time. We have a limit on the maximum waiting time. If the limit is reached and we are unable to schedule the task for some reason, we notify the users and ask them to try again.

Conclusion

We discussed the difference between an OS-level task scheduler and a data center-level task scheduler. We explained that the data center-level task scheduling needs a distributed solution because of multiple tenants and dispersed resources. We learned that the queue is a major building block of a task scheduler. We also used distributed queues where we can scale with an increasing number of tasks to utilize an increasing number of resources.



Non-functional requirements fulfilled by our task scheduler system