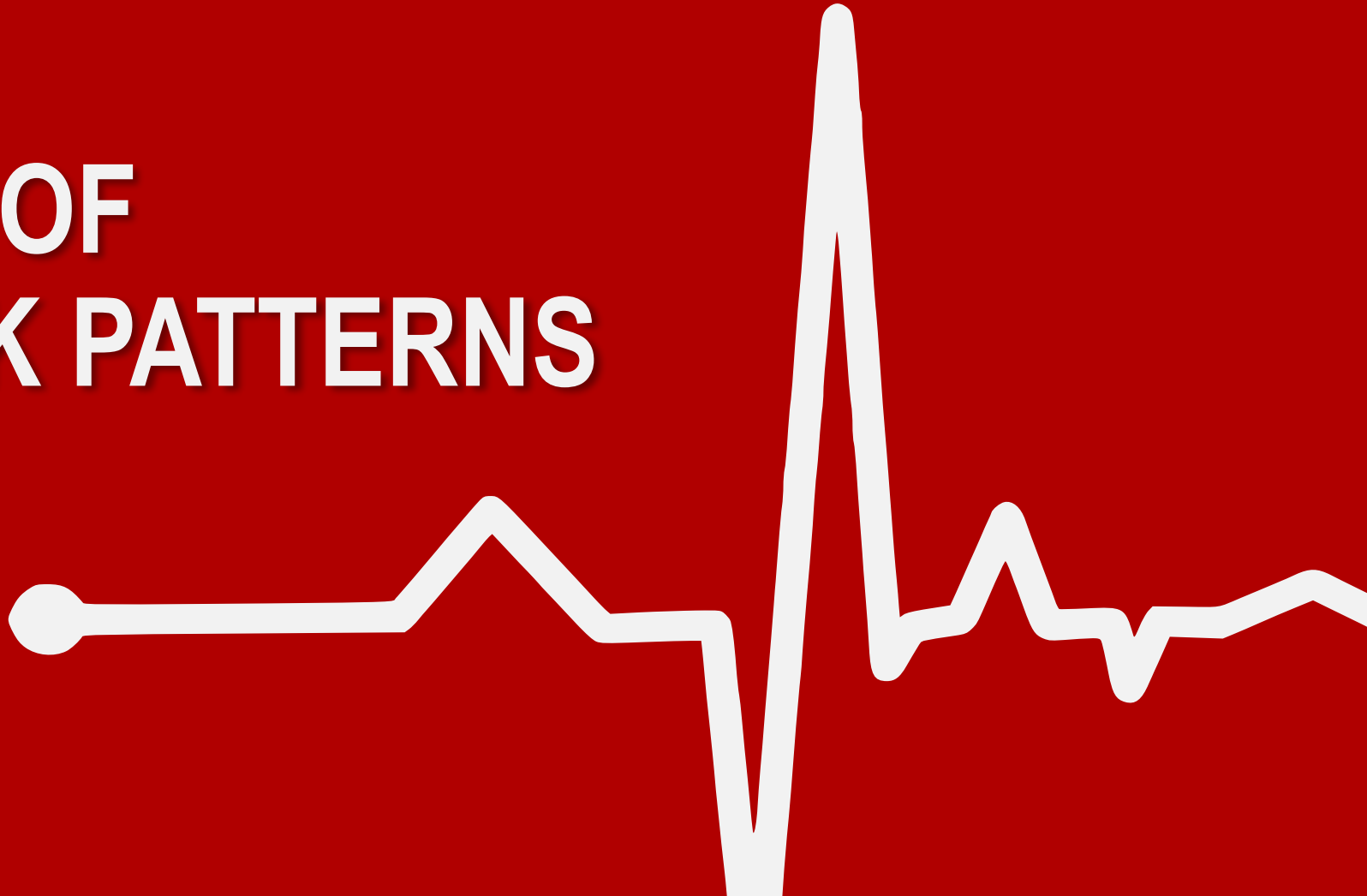
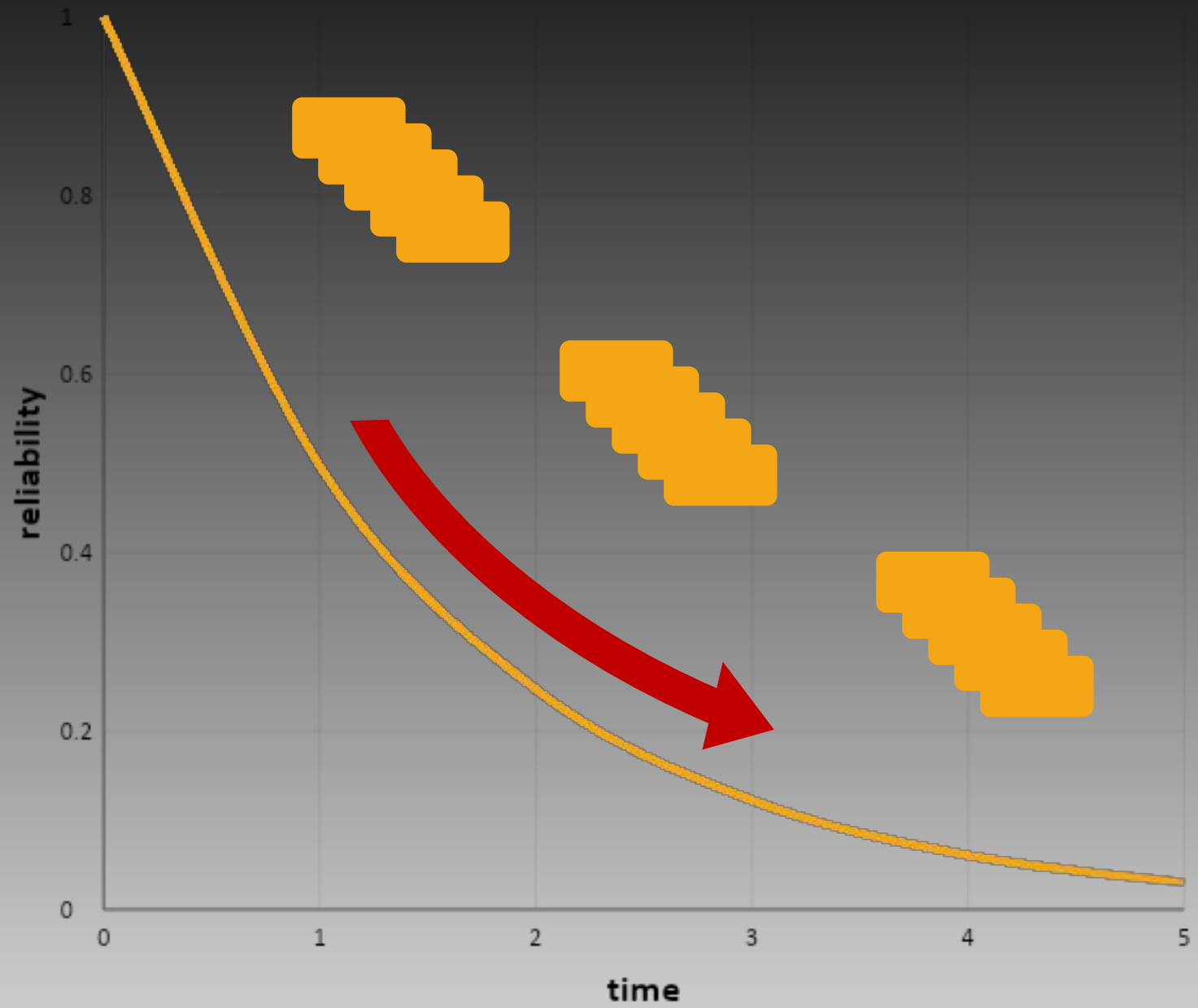


# AN OVERVIEW OF HEALTH CHECK PATTERNS







# 96%

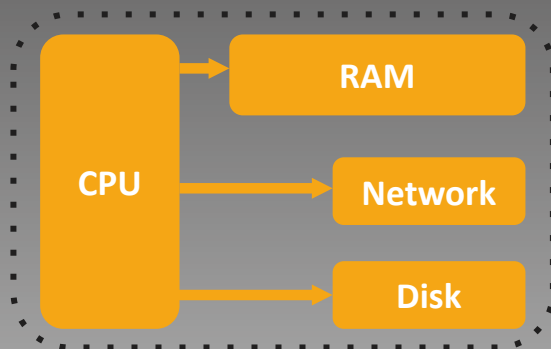


99%

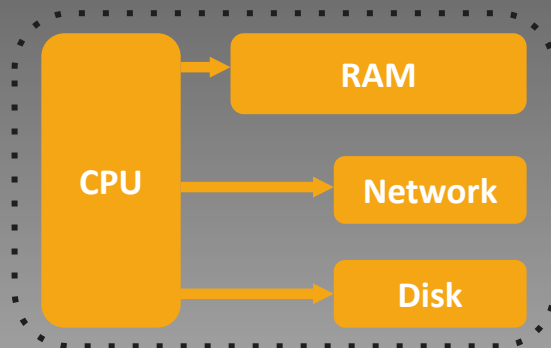
99%

99%

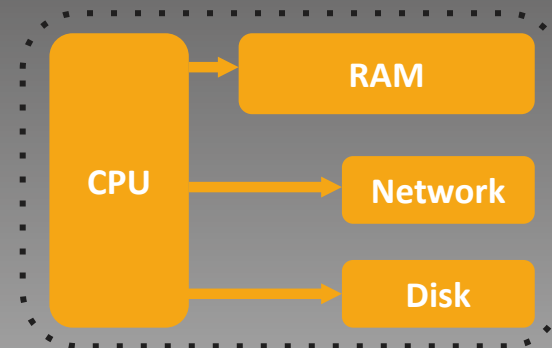
99%



**CLIENT**

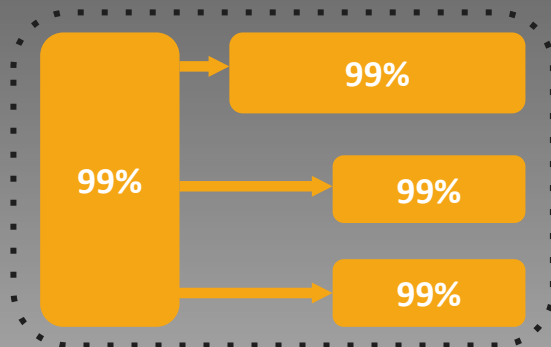


**SERVER**

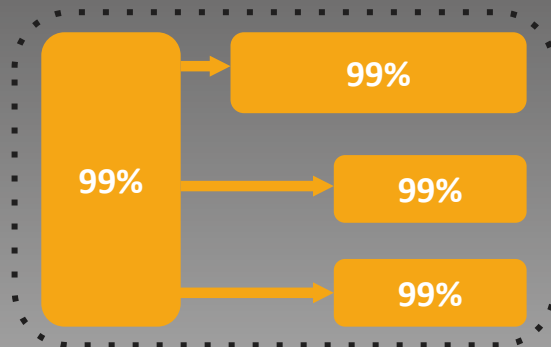


**DB**

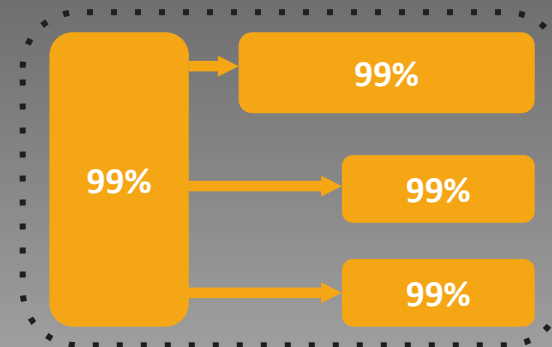
# 88.5%



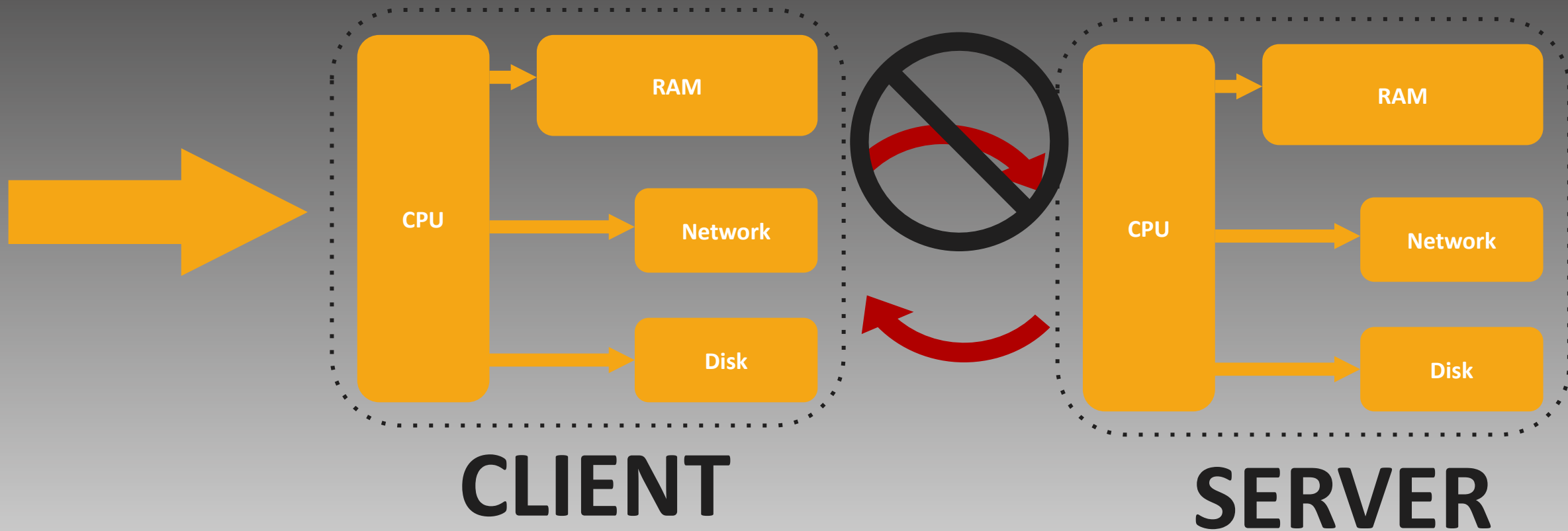
**CLIENT**



**SERVER**



**DB**

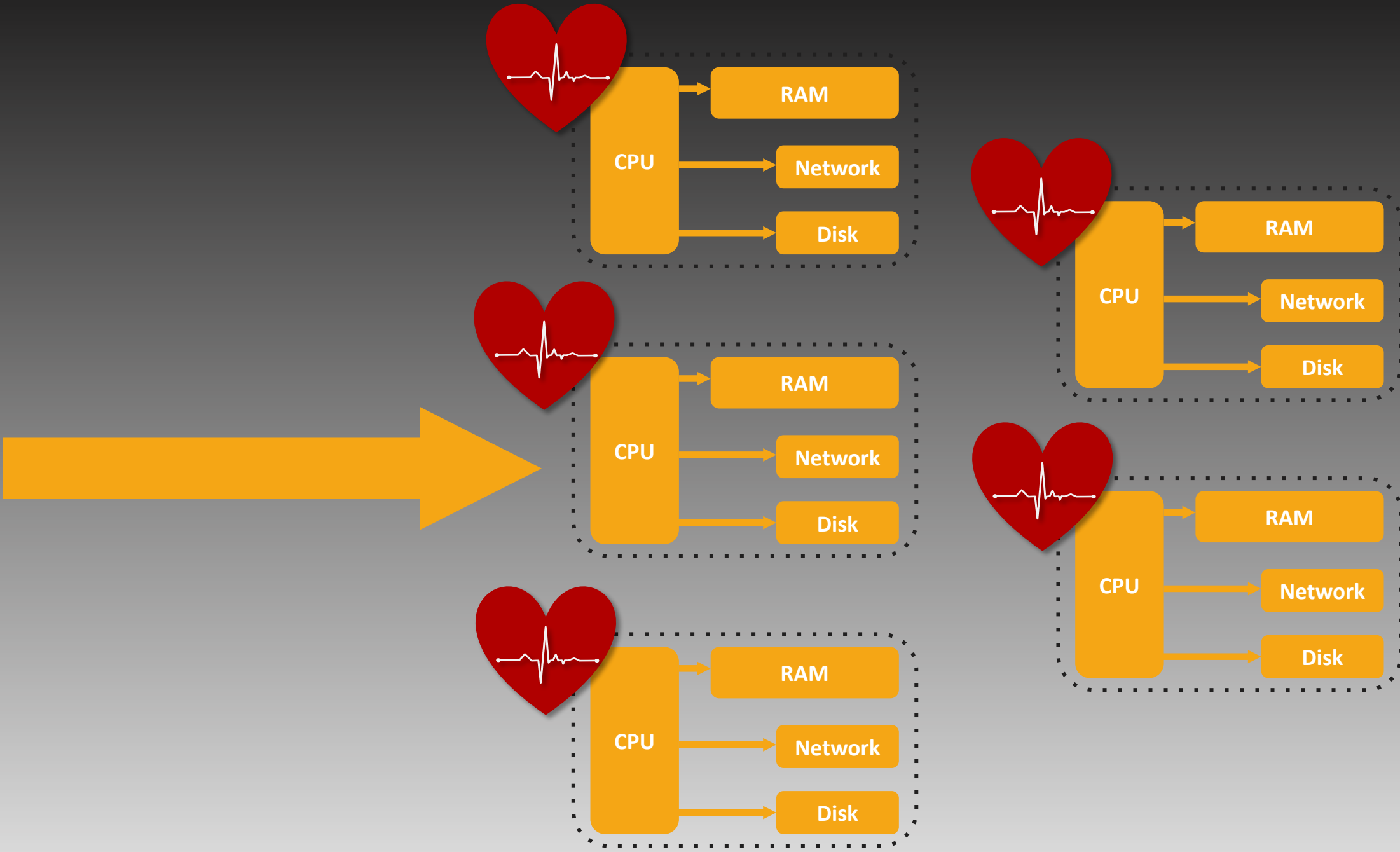


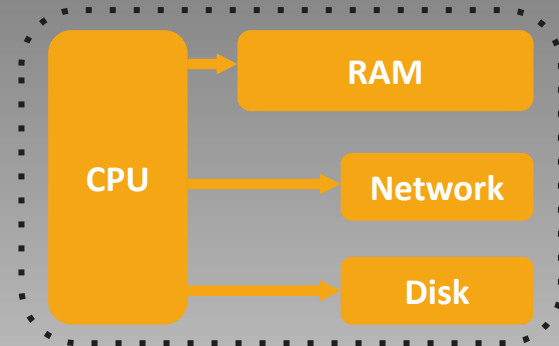
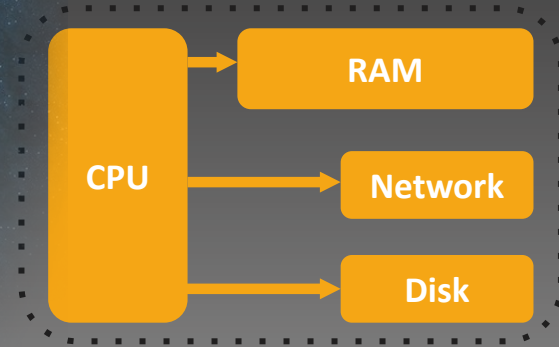
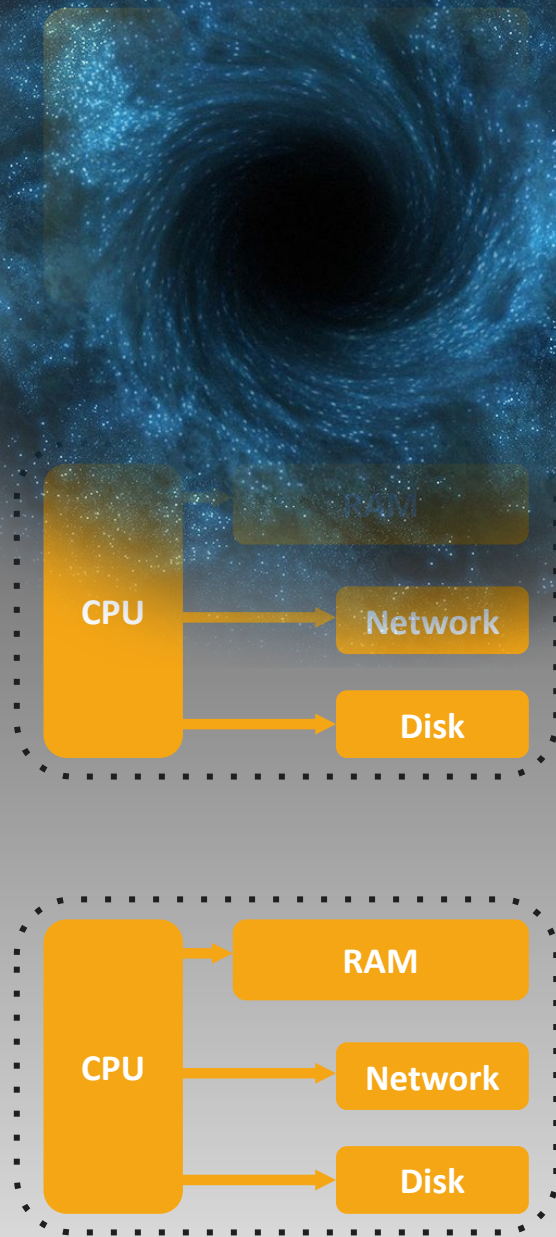


IT Team before going on Holiday

- × bugs
- × config issue
- × synchronization failed
- × memory leak
- × thread leak
- × pool misconfigurations
- × deadlocks

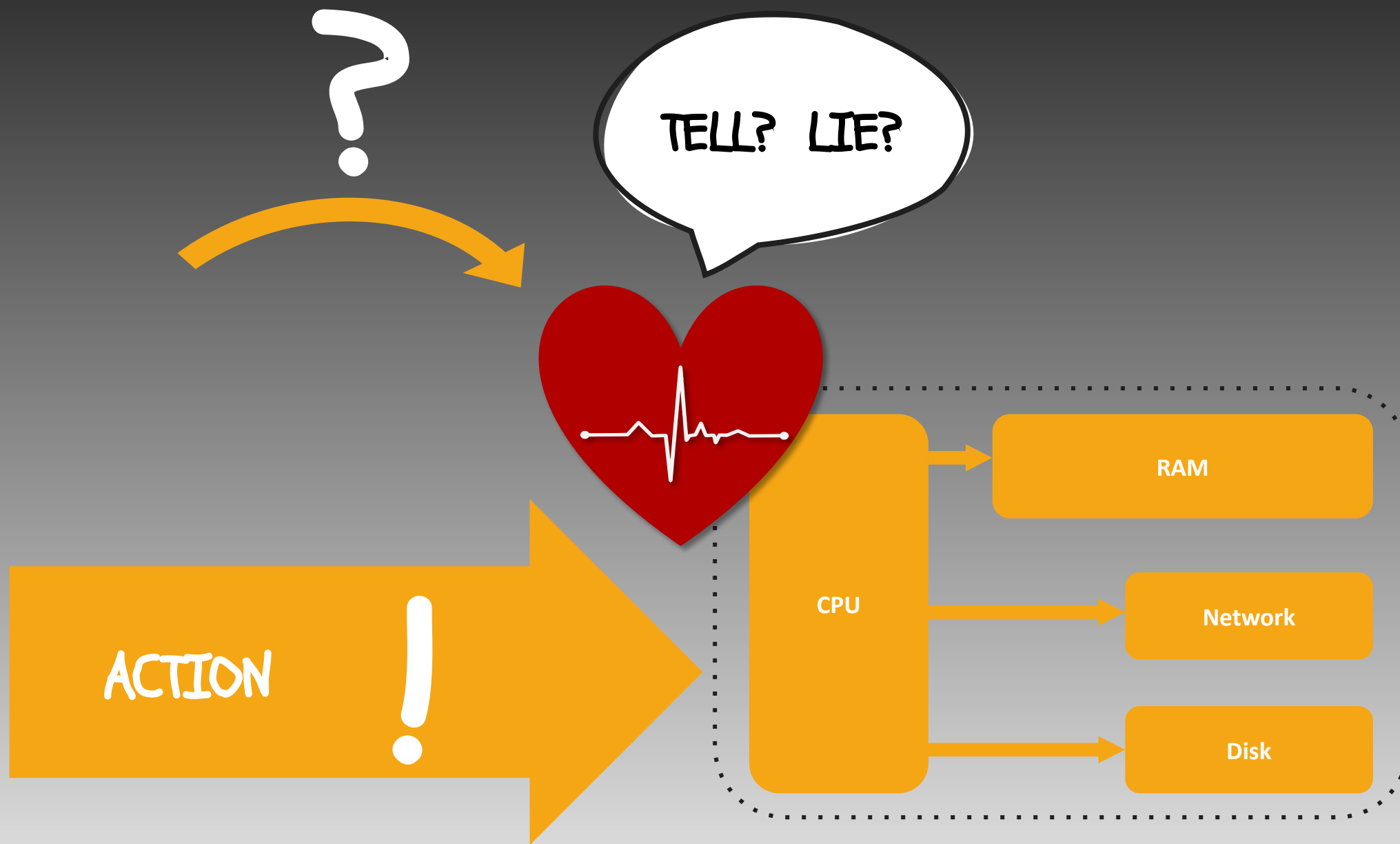






**HEALTH CHECK?**





**ACTION**



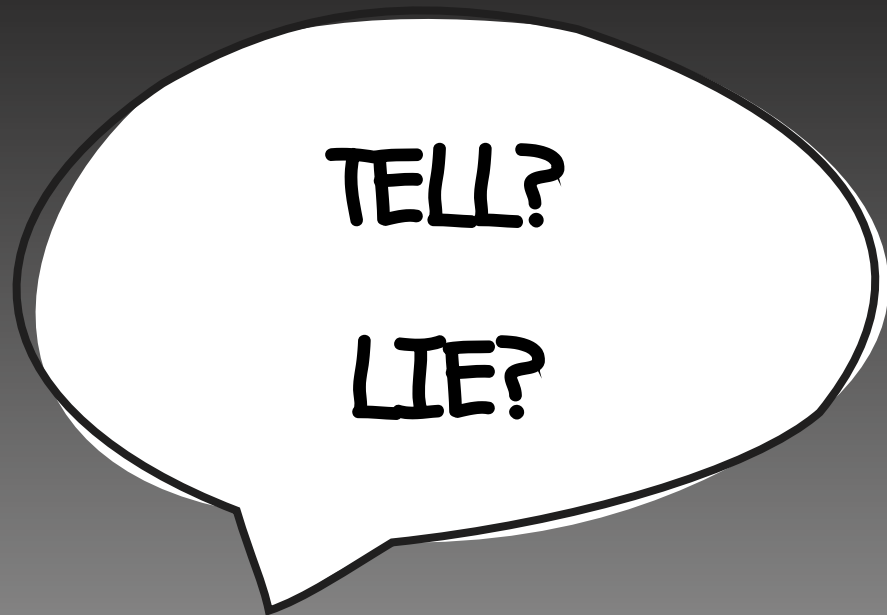
**RESTART**

**ALERTING**

**TRAFFIC  
SHAPING**

**SCALING**

**DEPLOYMENTS**

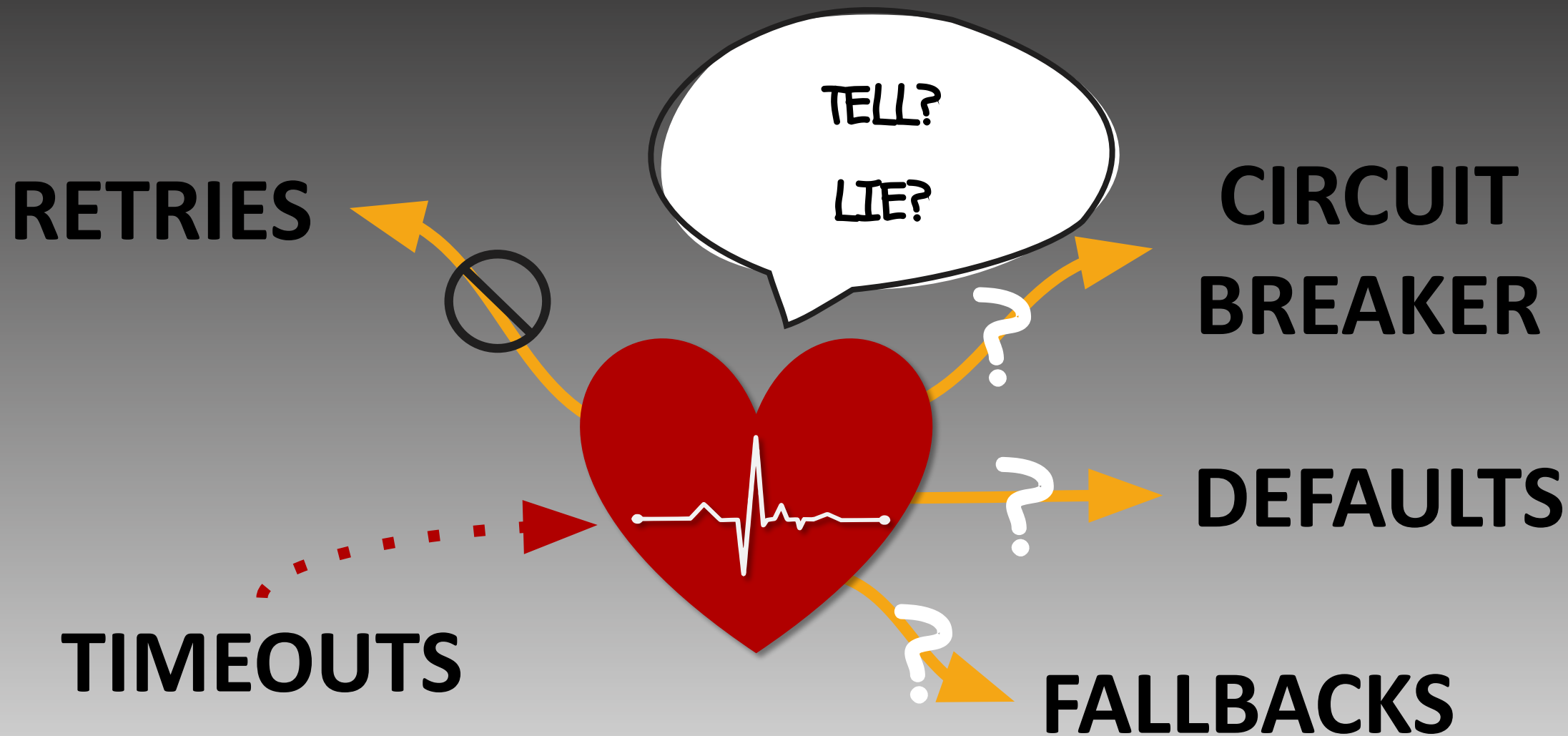


**SHALLOW**

**N/A**

**DEEP**

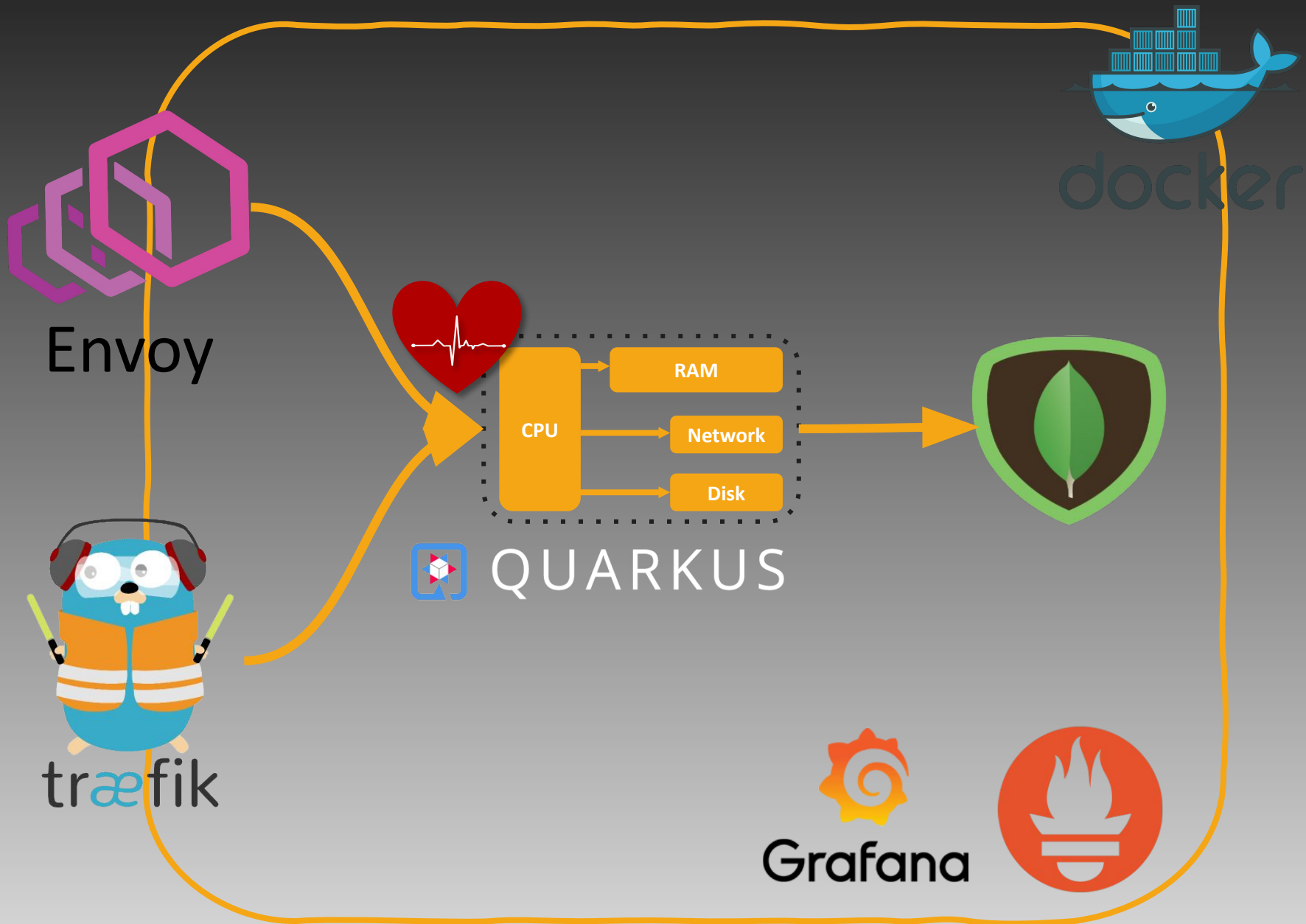
**PASSIVE**

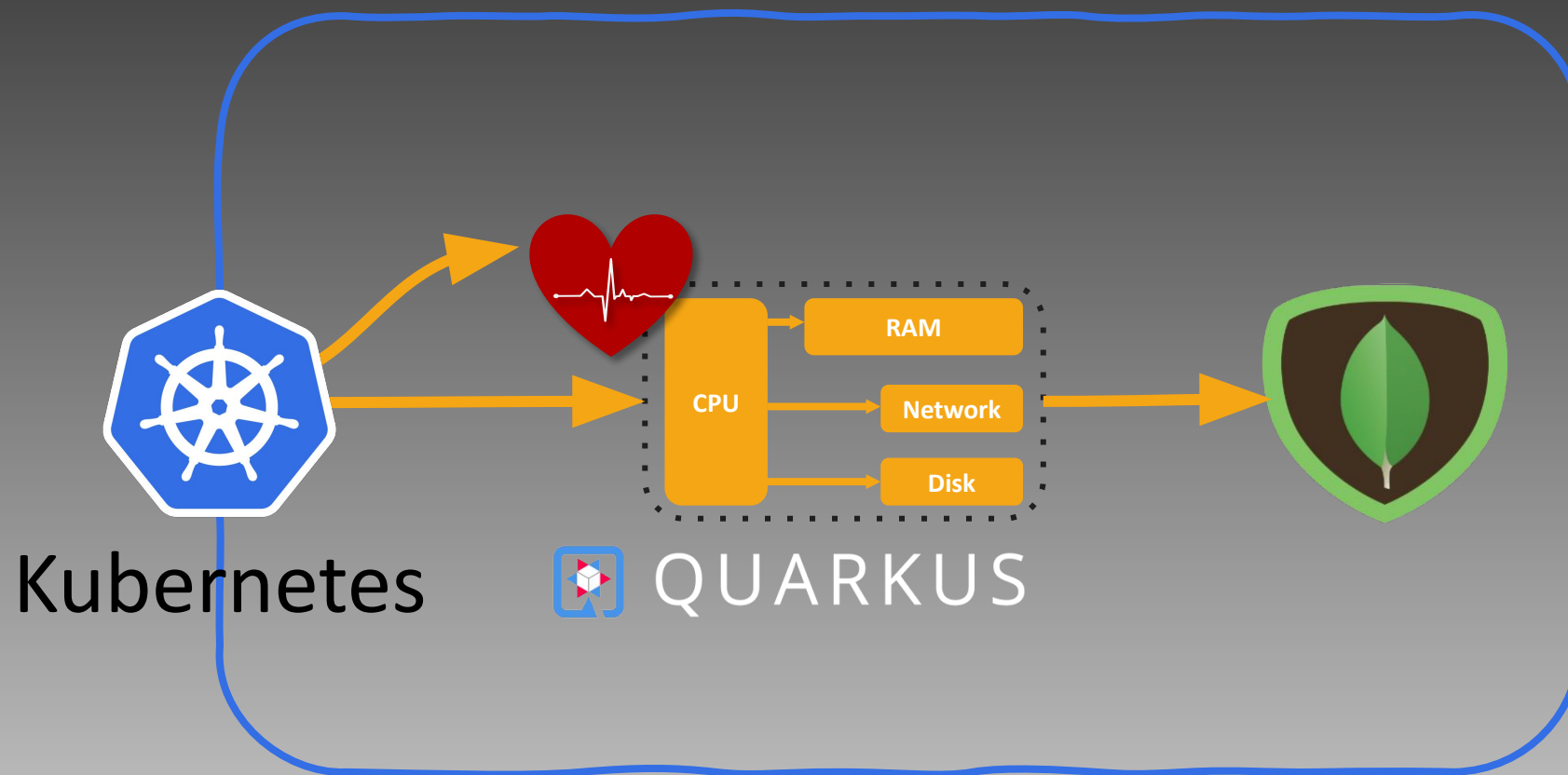


**PATTERNS!**











N/A

## RESTARTS

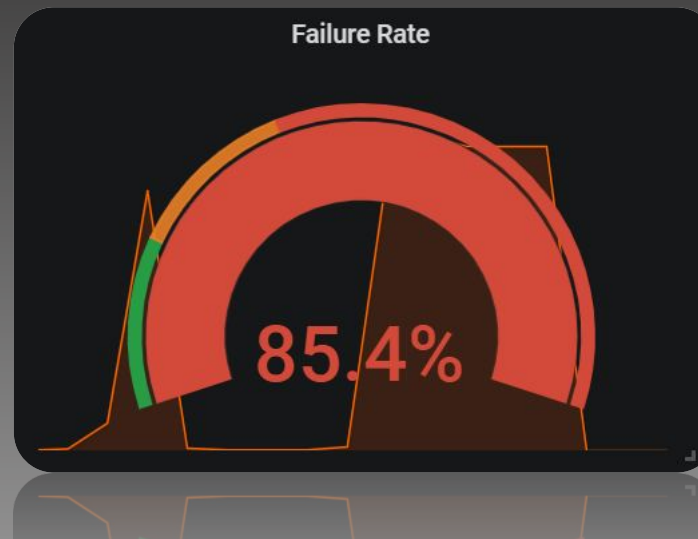
```
spec:
```

```
  replicas: 2
```

```
  restartPolicy: Never
```

N/A

## ALERTING



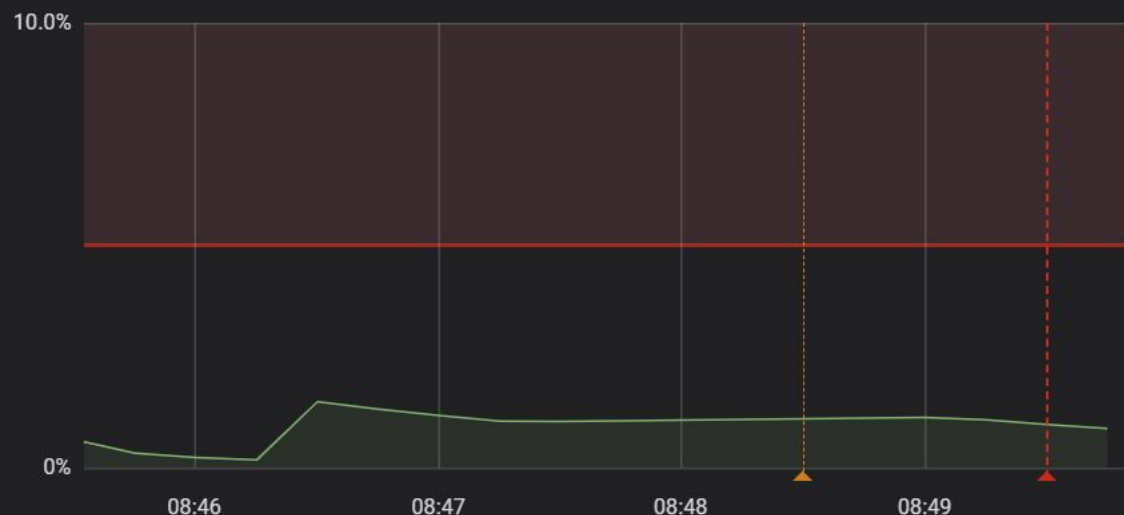
```
sum(rate(traefik_entrypoint_requests_total{code=~"5.."}[$interval])) /  
sum(rate(traefik_entrypoint_requests_total[$interval]))|
```

```
sum(rate(envoy_cluster_upstream_rq_xx{envoy_response_code_class="5"}[$interval])) /  
sum(rate(envoy_cluster_upstream_rq_xx[$interval]))|
```

N/A

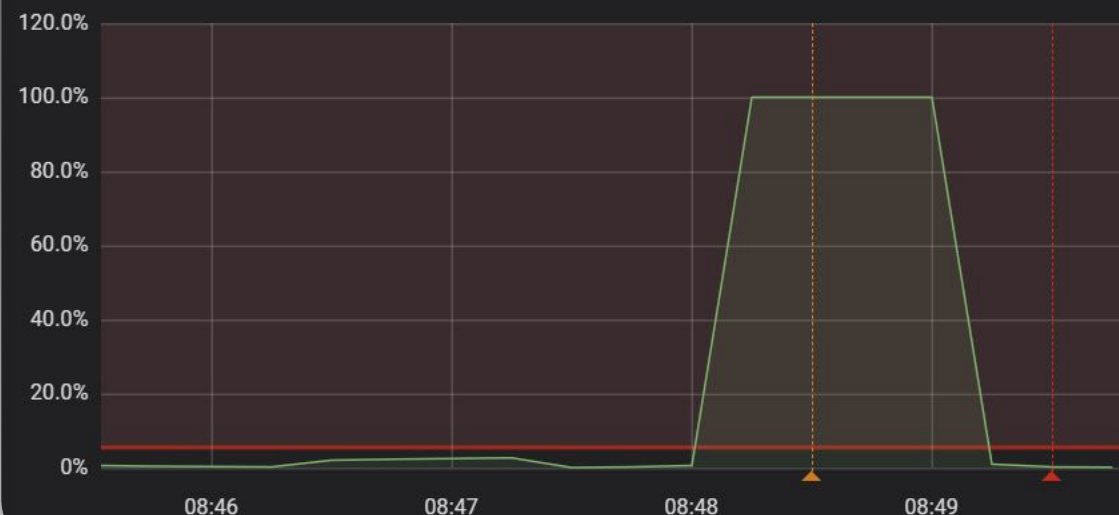
## ALERTING

♥ Failure Rate Alert



Interval = 5m

♥ Failure Rate Alert



Interval = 1m



**TRAFFIC SHAPING  
DEPLOYMENTS**



**N/A**

- ✓ memory leak
- ✓ thread leak
- × bugs
- × config issue
- × pool misconfigurations
- × deadlocks

**SHALLOW**

```
{  
  "status": "UP",  
  "checks": [  
    {  
      "name": "disk",  
      "status": "UP",  
      "data": {  
        "usable bytes": 39481786368  
      }  
    }  
  ]  
}
```



SHALLOW

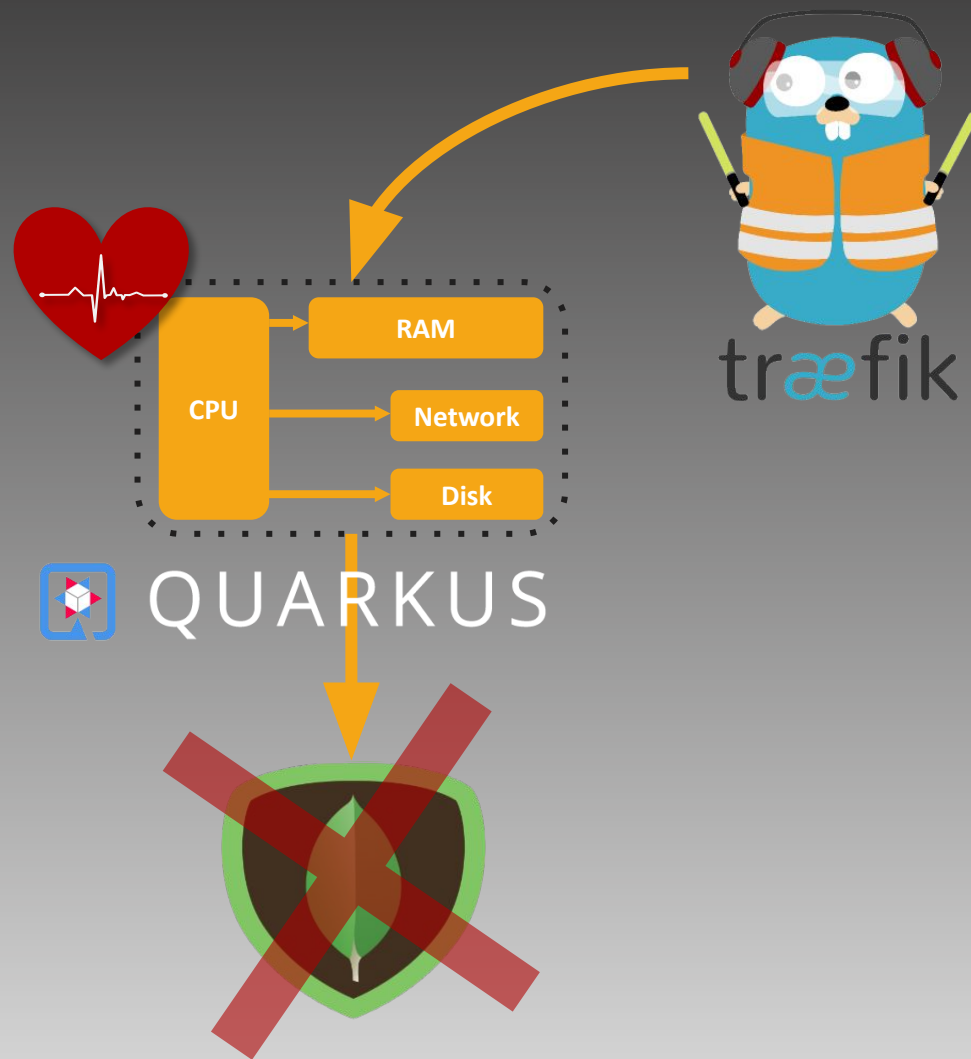
## RESTARTS

```
spec:  
  replicas: 2
```

```
livenessProbe:  
  httpGet:  
    path: /application/health/live  
    port: http  
  failureThreshold: 2  
  initialDelaySeconds: 3  
  periodSeconds: 3
```

SHALLOW

## TRAFFIC SHAPING





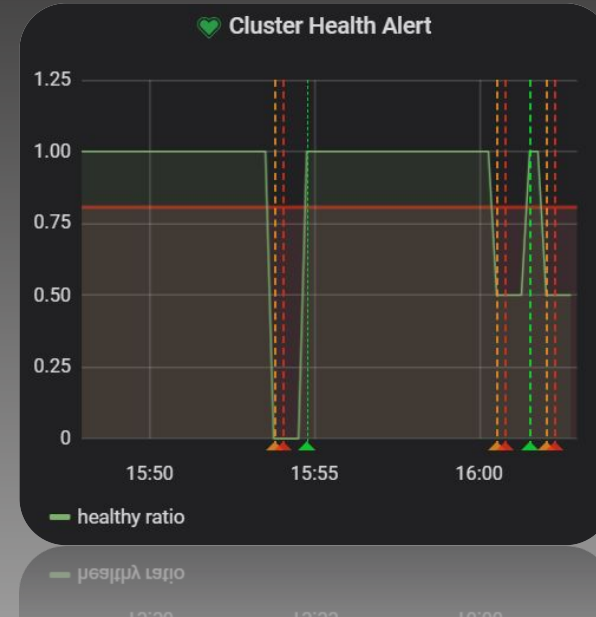
**SHALLOW**

## CONNECTION POOL ISSUES

- × timeout
- × invalid connections
- × blocked threads
- × unbounded pool
- × overflowing
- × thread / memory leaks

SHALLOW

## ALERTING



```
sum(traeфик_service_server_up{service="application@file"}) /  
sum(max_over_time(traeфик_service_server_up{service="application@file"}[10m]))
```

```
envoy_cluster_health_check_healthy{envoy_cluster_name="application"} /  
max_over_time(envoy_cluster_membership_total{envoy_cluster_name="application"}[1d])
```



**SHALLOW**

**DEPLOYMENTS**





# SHALLOW

- ✓ memory leak
- ✓ thread leak
- × bugs
- × config issue
- × pool misconfigurations
- × deadlocks



**DEEP**

```
{  
  "status": "UP",  
  "checks": [  
    {  
      "name": "MongoDB connection health check",  
      "status": "UP",  
      "data": {  
        "default": "admin, config, hello, local"  
      }  
    }  
  ]  
}
```



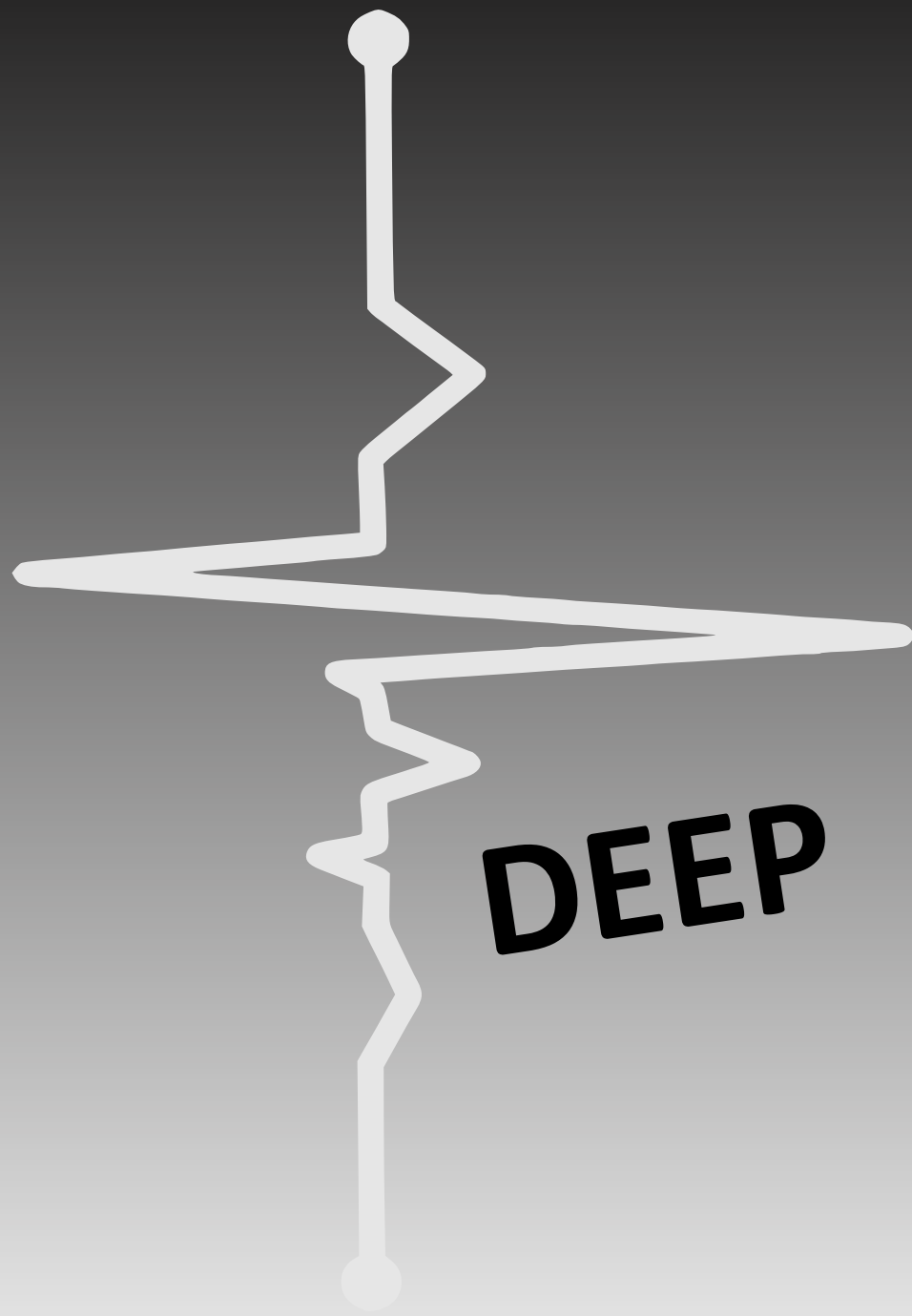
## RESTARTS & DEPLOYMENTS

```
livenessProbe:  
  httpGet:  
readinessProbe:  
  httpGet:  
startupProbe:  
  httpGet:  
    path: /application/health  
    port: http  
failureThreshold: 3  
periodSeconds: 5
```

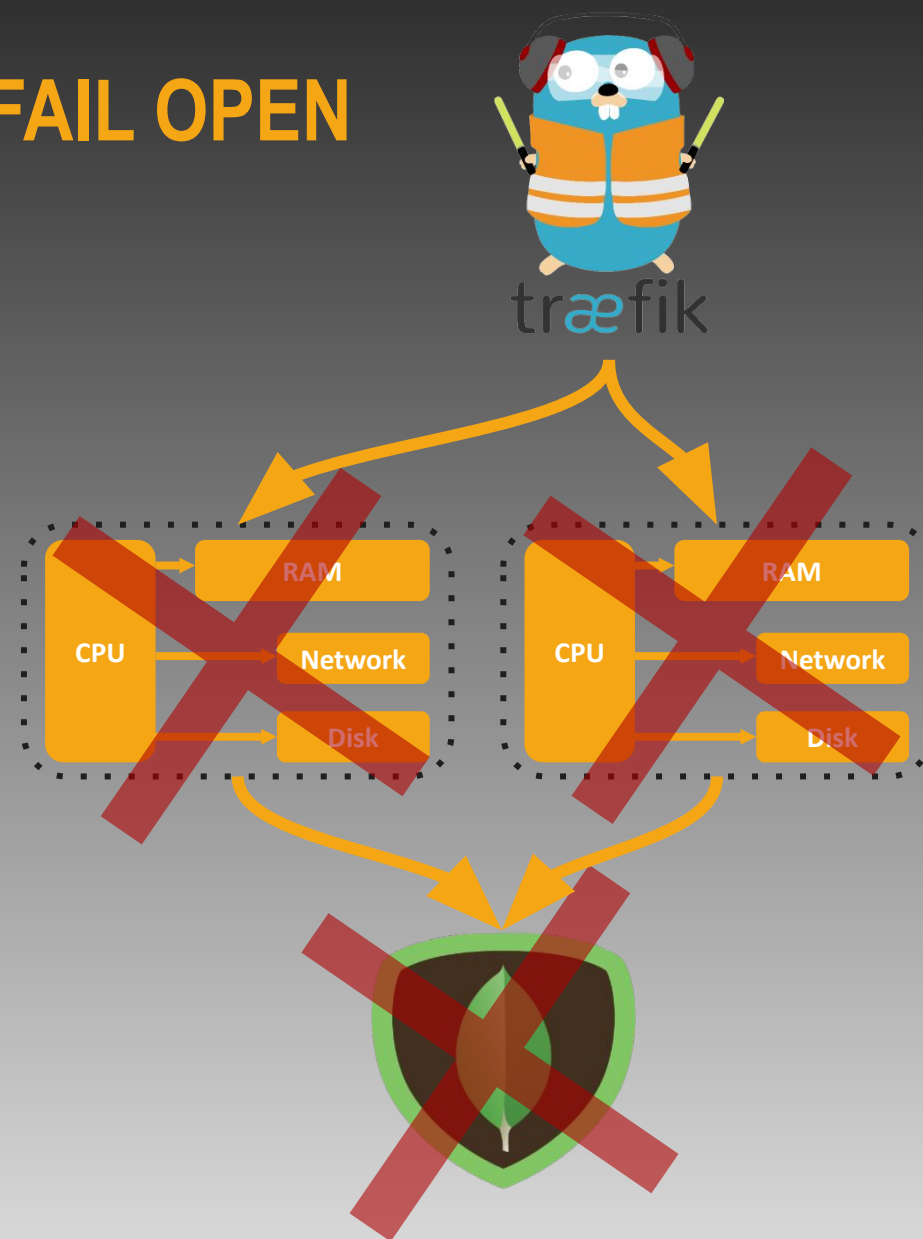
```
backoffSeconds: 2
```

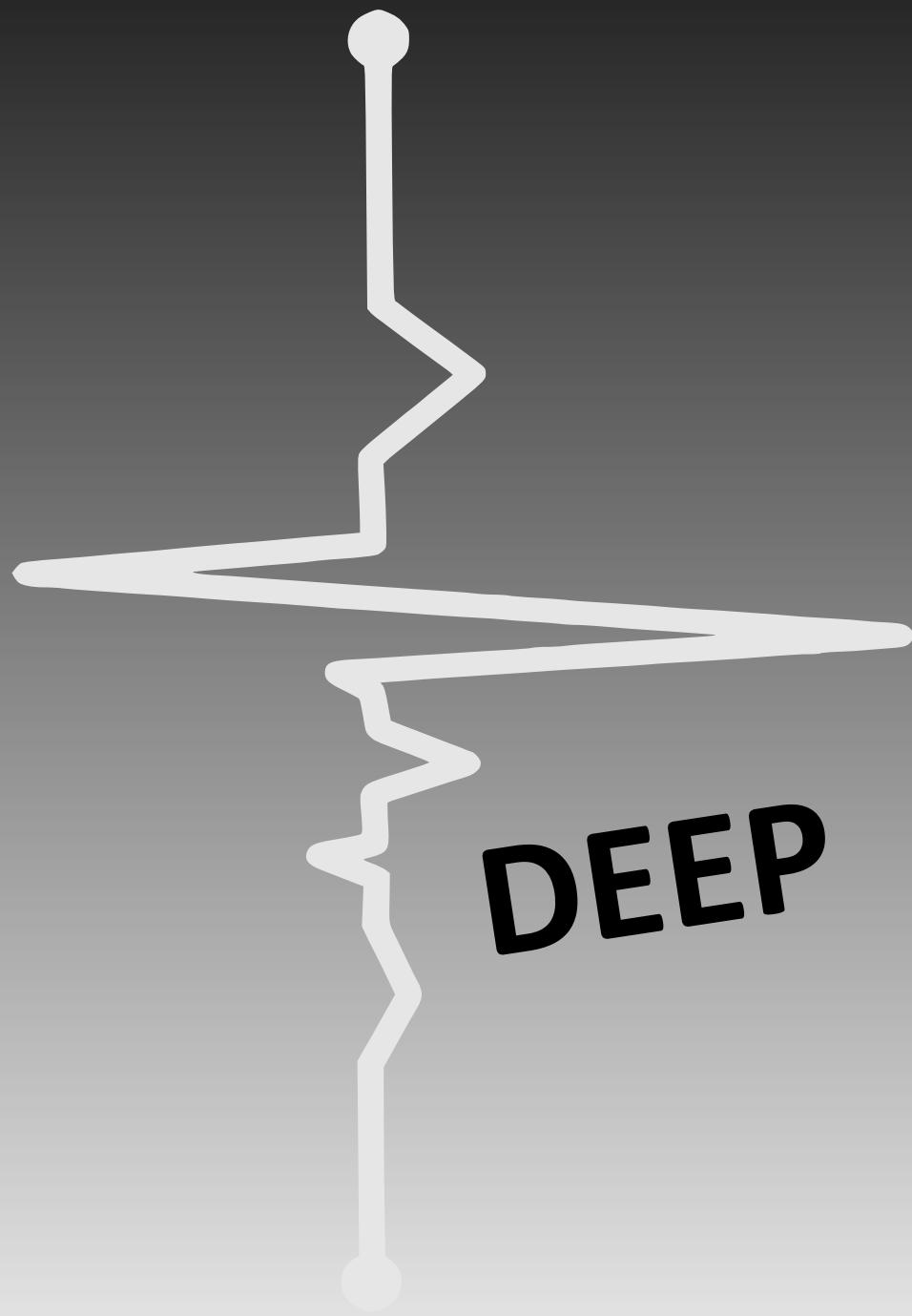
```
restartPolicy: Always
```



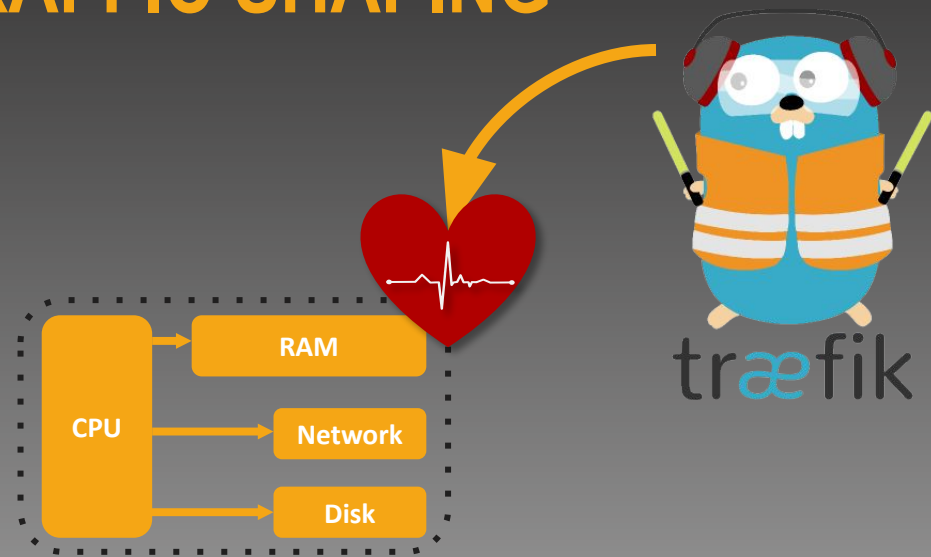


**FAIL OPEN**

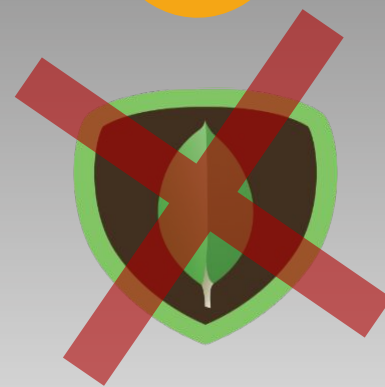


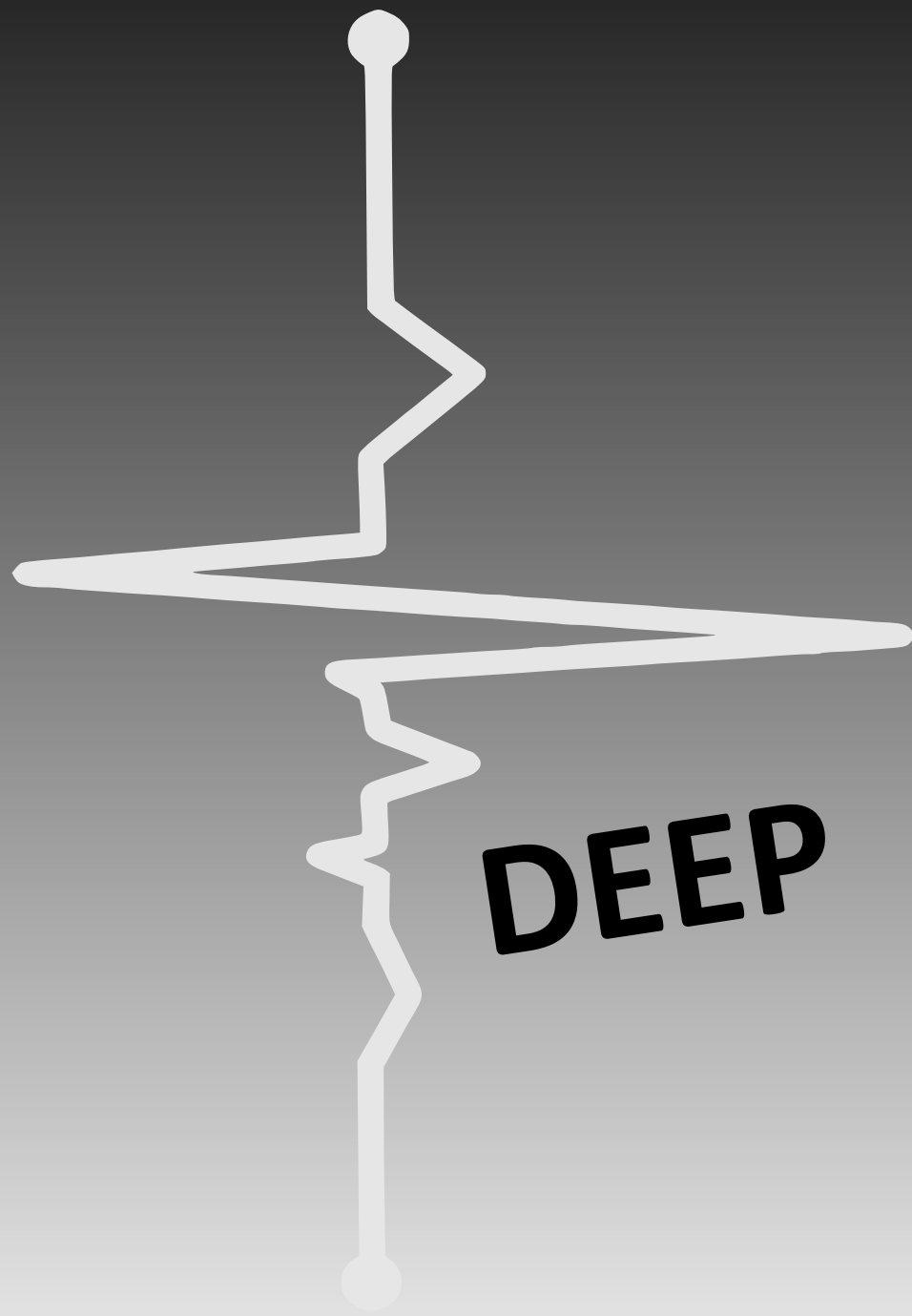


## TRAFFIC SHAPING

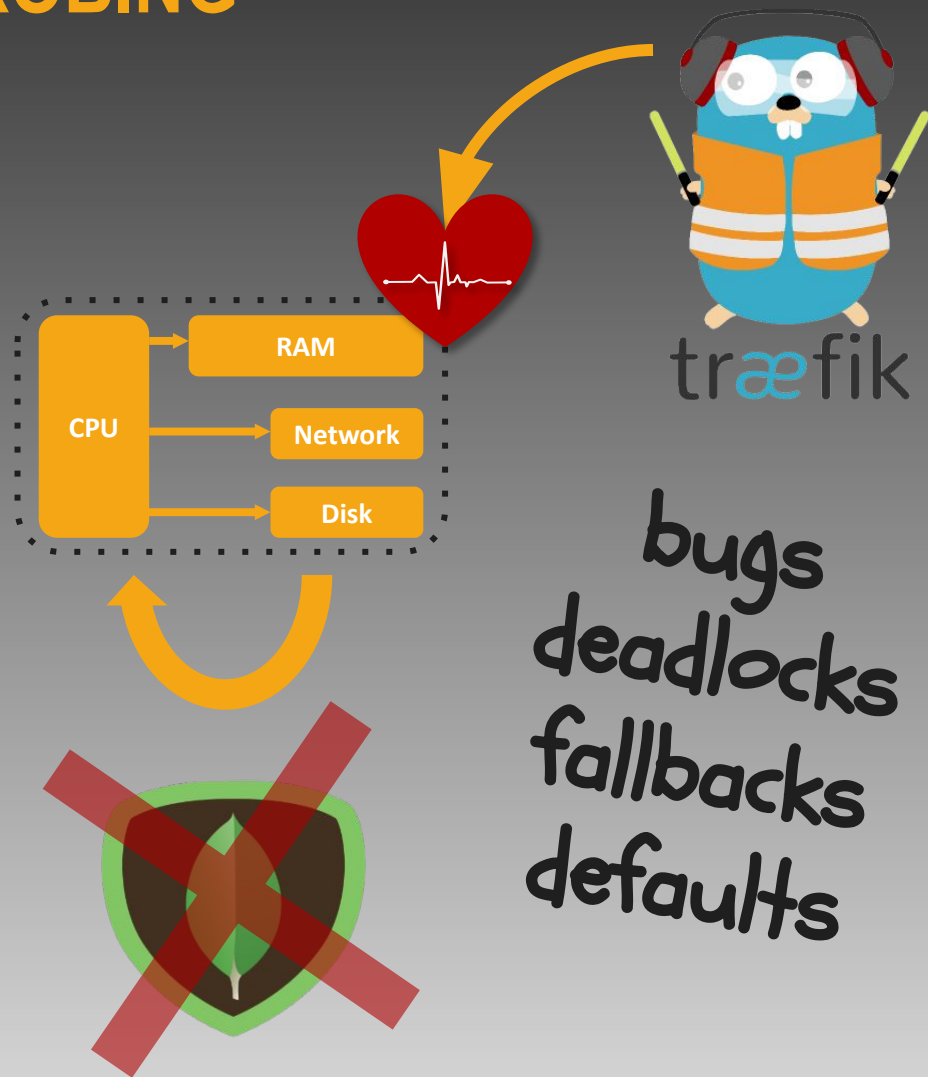


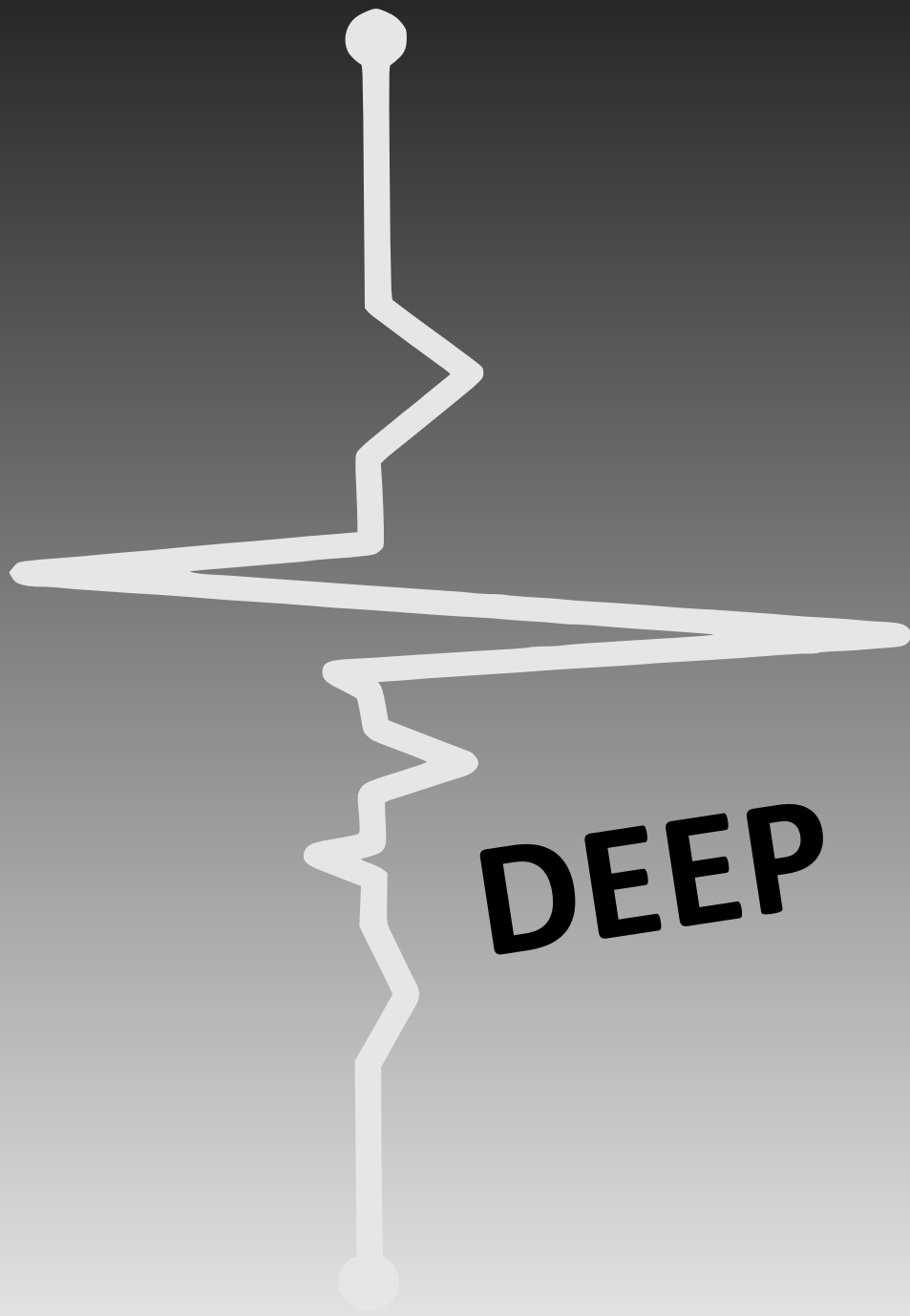
fallbacks  
defaults ?





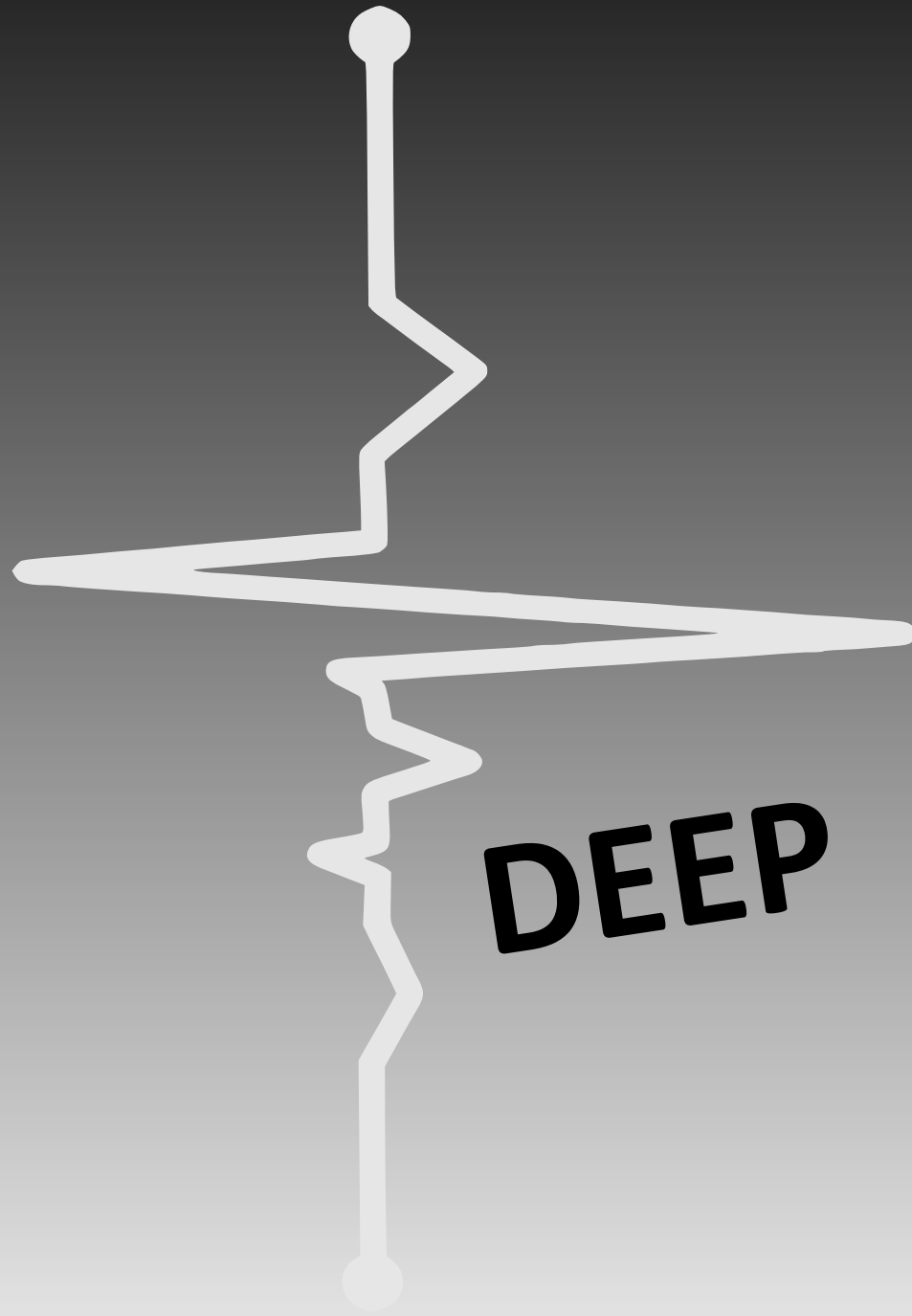
## PROBING





## PROBING

```
{  
  "status": "UP",  
  "checks": [  
    {  
      "name": "Probing health check",  
      "status": "UP",  
      "data": {  
        "Successful transactions": 3,  
        "enabled": true  
      }  
    }  
  ]  
}
```



- ✓ memory leak
- ✓ thread leak
- ✓ pool misconfigurations
- ✓ bugs
- ✓ config issue
- ✓ deadlocks



**PASSIVE**

```
{  
  "status": "UP",  
  "checks": [  
    {  
      "name": "Metering health check",  
      "status": "UP",  
      "data": {  
        "last call since(ms)": 4434,  
        "failed ratio": "0.0",  
        "enabled": true  
      }  
    }  
  ]  
}
```



**PASSIVE**

## TRAFFIC SHAPING

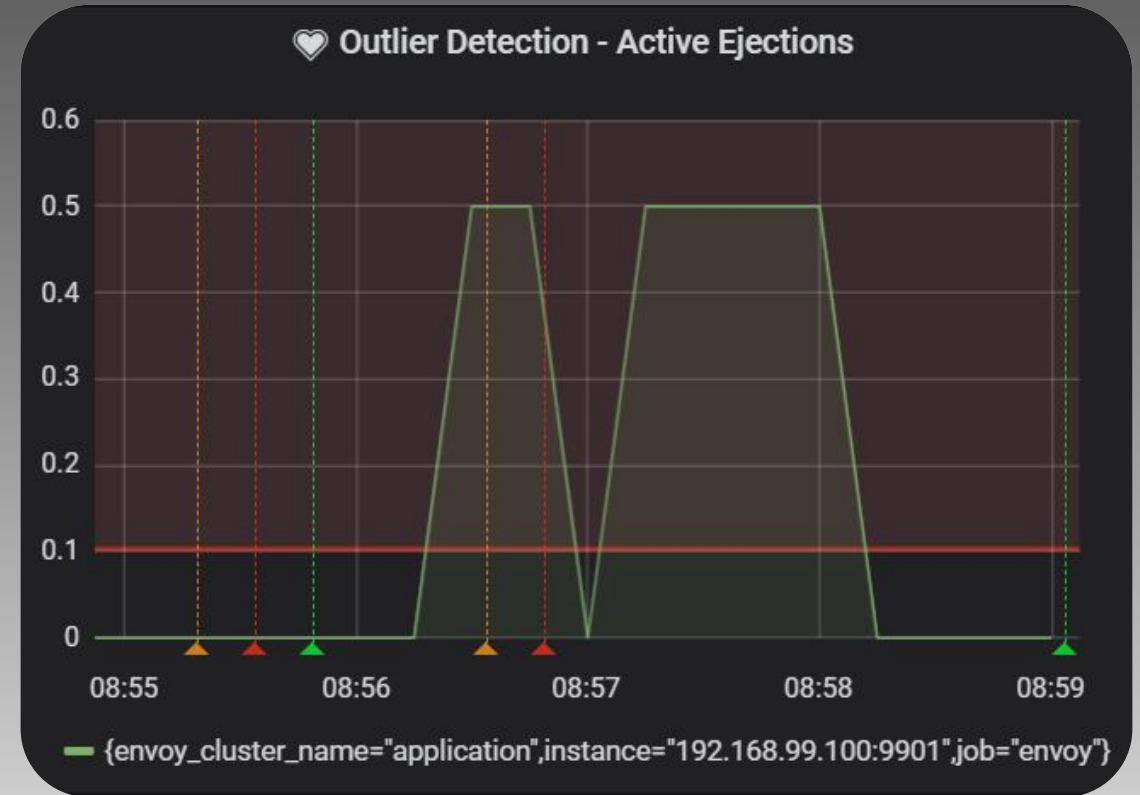
```
clusters:  
- name: application  
  connect_timeout: 0.25s  
  type: strict_dns  
  lb_policy: round_robin  
  http2_protocol_options: {}  
  outlier_detection:  
    consecutive_5xx: 3  
    interval: 5s  
    base_ejection_time: 30s  
    max_ejection_percent: 50
```

```
max_ejection_percent: 20  
base_ejection_time: 30s
```

# ALERTING

```
envoy_cluster_outlier_detection_ejections_active{envoy_cluster_name="application"} /  
max_over_time(envoy_cluster_membership_total{envoy_cluster_name="application"}[1d])
```

PASSIVE







- ✓ memory leak
- ✓ thread leak
- ✓ pool misconfigurations
- ✓ bugs
- ✓ config issue
- ✓ deadlocks

**PASSIVE**

**DEMO!**



# THANK YOU!



## ORESZTÉSZ MARGARITISZ

# <epam>



@gitaroktat

o



gitaroktat

o



<https://www.linkedin.com/in/oresztasz>

# LINKS & REFERENCES

<https://github.com/gitaroktato/healthcheck-patterns>

<https://github.com/gitaroktato/microservices-availability-simulator>



# AN OVERVIEW ON HEALTH CHECK PATTERNS

