# MANAGING SERVICE RELIABILITY BY MANAGING RISKS

# PRESENTER

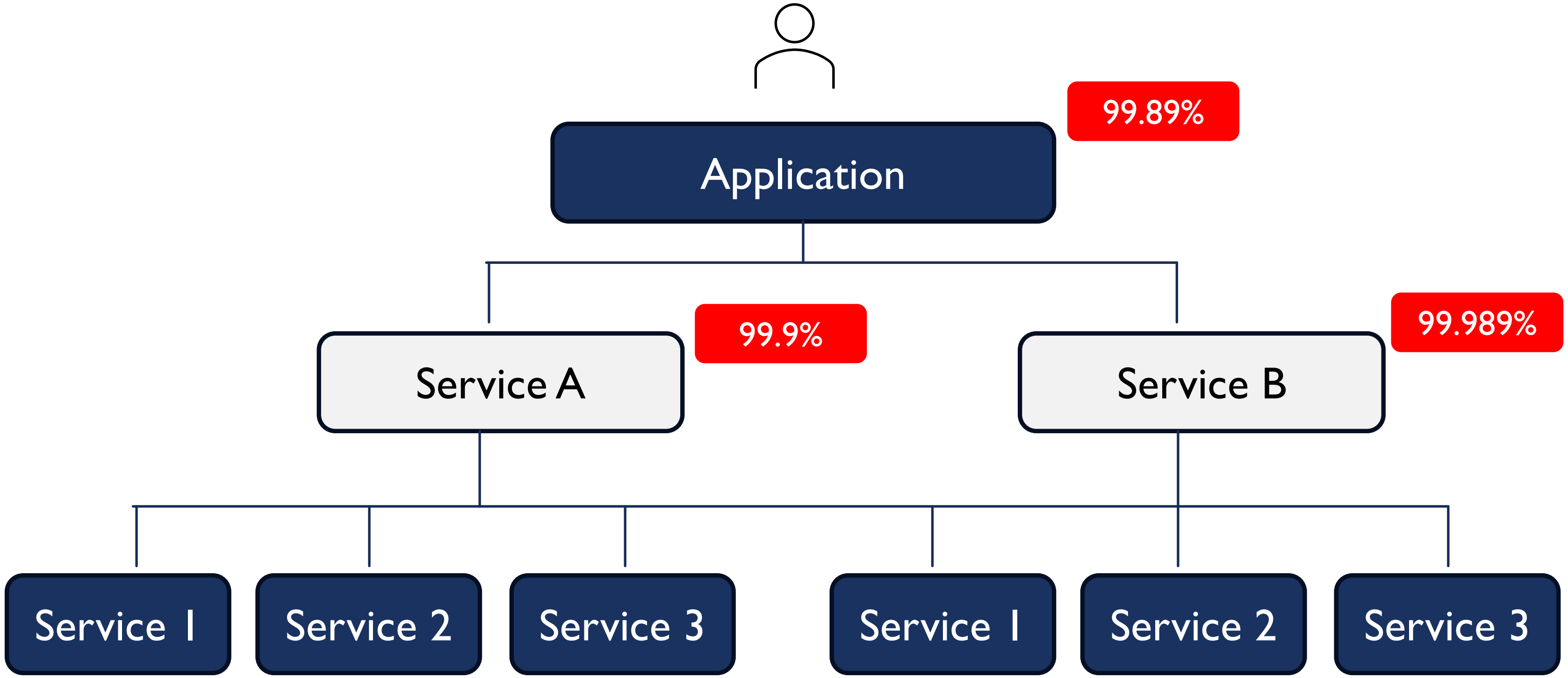
Jaiprakash Narain Pherwani

Center of Excellence Lead – Technology Practices

Tata Consultancy Services (TCS)

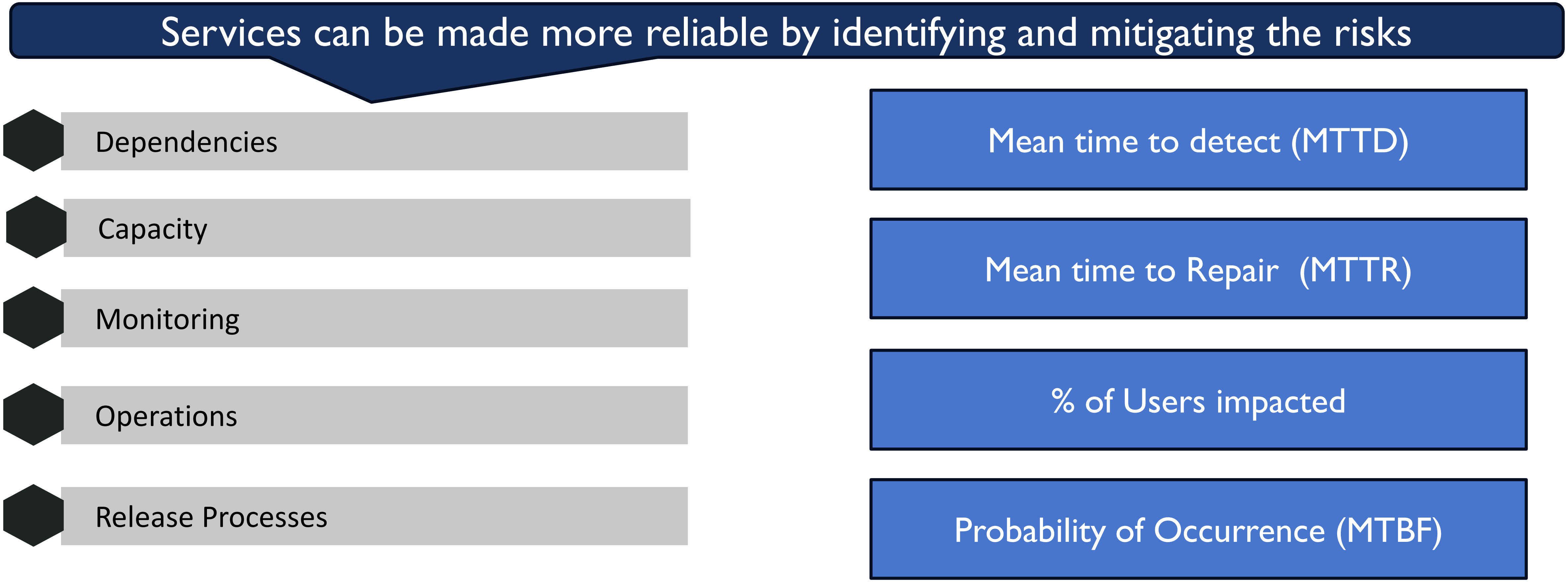
# SLO'S – REALISTIC ?

- SLO's are foundation tasks for the SRE practices
  - Detection and alerting in place ?
  - Is your application Architecture able to deliver the response and the user experience
  - How about Infrastructure resiliency ?
  - Auto healing in place ?
  - New releases, if it impacts, do you have options to roll back ?



# RISK ANALYSIS

- Prioritize and communicate risks to the given Service



Services can be made more reliable by identifying and mitigating the risks

- Dependencies
- Capacity
- Monitoring
- Operations
- Release Processes

- Mean time to detect (MTTD)
- Mean time to Repair (MTTR)
- % of Users impacted
- Probability of Occurrence (MTBF)

# RISK CATALOG

**Risk catalog is a structured way to capture all the risks for each of the Service, it enables to devise and prioritize the risks  based on Various Risk factors**

## Defining your Risk Catalog

Infrastructure and Software

Include SLI, MTTD, MTTR & MTBF

Customer User Journey

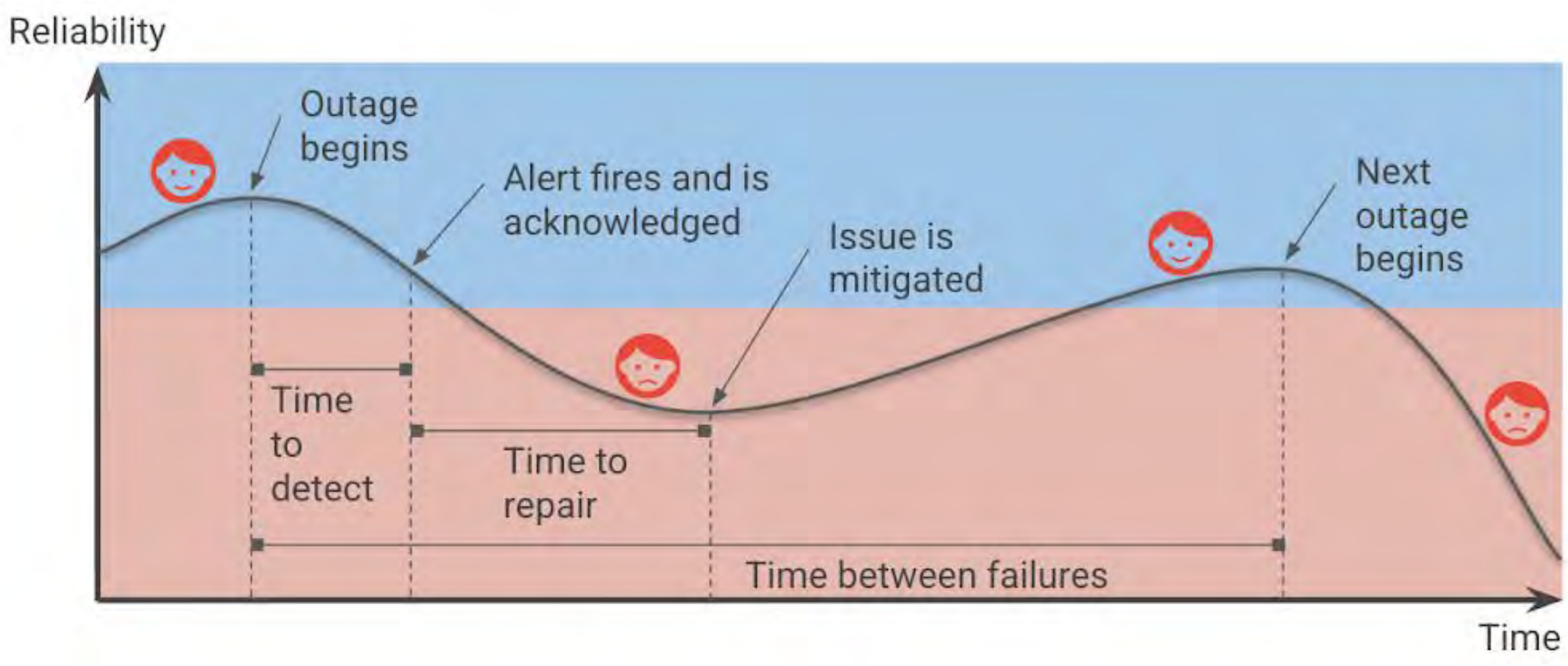
Refer Past Incidents

# TYPICAL RISK CATALOG

| | (in minutes) | | | (in days) | | |
|---|---|---|---|---|---|---|
| Risk | MTTD | MTTR | % Impact | MTTF | incidents/year | bad mins/year |
| Myself - A configuration mishap reduces capacity; causing overload and dropped requests | 30 | 120 | 20% | 120 | 3.04 | 91 |
| Myself - A new release breaks a small set of requests; not detected for a day; quick rollback when detected. | 1440 | 40 | 2% | 90 | 4.06 | 120 |
| Myself - A new release breaks a sizeable subset of requests; unfamiliar rollback procedure extends outage | 5 | 120 | 50% | 180 | 2.03 | 127 |
| Users - Unnoticed growth in usage triggers overload; service collapses. | 5 | 60 | 100% | 365 | 1.00 | 65 |
| Myself - Operator is slow to debug and root cause bug due to noisy alerting | 240 | 10 | 15% | 180 | 2.03 | 76 |
| Myself - Operator accidentally deletes database; restore from backup is required | 5 | 510 | 100% | 1461 | 0.25 | 129 |
| Dependency - Provision for Cloud provider single-zone VM/network outage | 5 | 40 | 50% | 365 | 1.00 | 23 |
| Dependency - Provision for Cloud provider regional VM/network outage | 2 | 30 | 100% | 730 | 0.50 | 16 |
| | | | | | | 0 |
| | | | | | | 0 |

Source – Google SRE Playbook

# RATE YOUR RISKS



- Create the Risk analysis Catalog
- Collaboration with different teams to quantify the risks
- Start with Estimates
- Collect more data from Incidents and update these Estimates
- Iterate and update the estimates based on Incidents in Production

# ACCEPTING RISKS

| Computed Stack Rank of Risks | bad mins/year | accept risk? | accept risk? | accept risk? |
|---|---|---|---|---|
| Myself - Operator accidentally deletes database; restore from backup is required | 129 | 🟥 | y | 🟩 |
| Myself - A new release breaks a sizeable subset of requests; unfamiliar rollback procedure extends outage | 127 | 🟥 | y | 🟩 |
| Myself - A new release breaks a small set of requests; not detected for a day; quick rollback when detected. | 120 | 🟥 | y | 🟩 |
| Myself - A configuration mishap reduces capacity; causing overload and dropped requests | 91 | 🟥 | 🟨 | 🟩 |
| Myself - Operator is slow to debug and root cause bug due to noisy alerting | 76 | 🟥 | 🟨 | 🟩 |
| Users - Unnoticed growth in usage triggers overload; service collapses. | 65 | 🟩 | 🟩 | 🟩 |
| Dependency - Provision for Cloud provider single-zone VM/network outage | 23 | 🟩 | 🟩 | 🟩 |
| Dependency - Provision for Cloud provider regional VM/network outage | 16 | 🟩 | 🟩 | 🟩 |

## Risk cannot be accepted

This risk is unacceptable, as it falls above the acceptable error budget for a single risk, and therefore, can have a major impact on your reliability in a single event.

## Risk Should not be accepted

This risk should not be acceptable, as it's a major consumer of your error budget and therefore, needs to be addressed. You may be able to accept some amber risks by addressing some less urgent (green) risks to buy back budget.

## Risk Could be accepted

This is an acceptable risk. It is not a major consumer of your error budget, and in aggregate, does not cause your application to exceed the error budget. You don't have to address green risks, but may wish to do so to give yourself more budget to cover unexpected risks, or to accept amber risks that are hard to mitigate or eliminate.

- Evaluate your SLO's , is it achievable given the risks
- How do you achieve the Targets
- Prioritize Engineering work that mitigates or eliminates any unacceptable risks.

# LEVERAGE CHAOS ENGINEERING

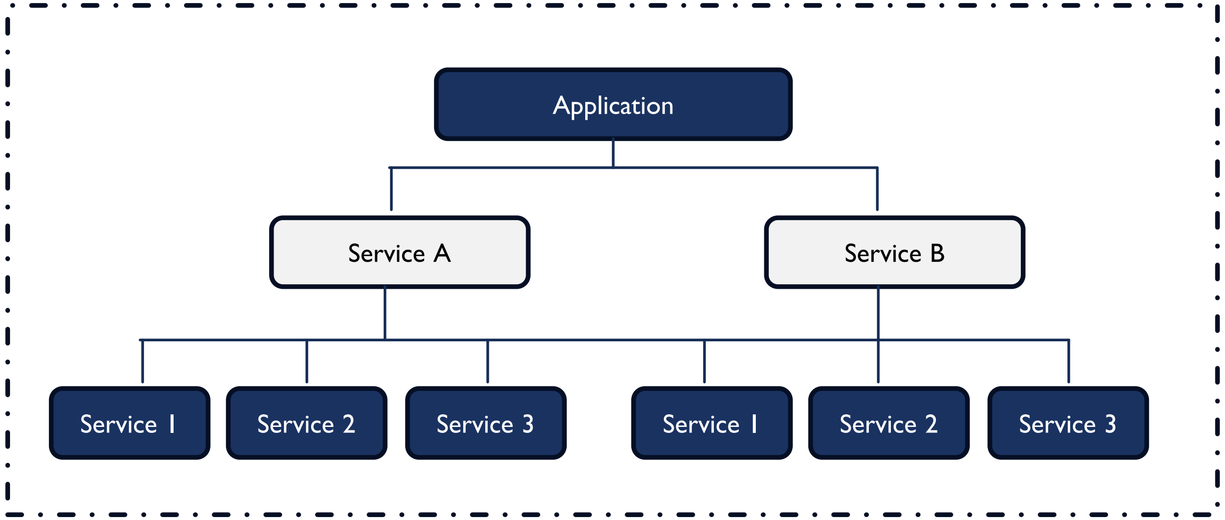
Chaos Engineering is a approach to proactively devise different fault scenario's and thereby identify blind spots, response of the systems etc.

- Estimate different risks with Chaos Engineering
- Refine your Estimated MTTD, MTTR Values
- Understand different Blind spots which go unnoticed and thereby include them in the Risk Catalog
- Right align your SLO's and  Prioritize Engineering Work

Infra Failures

Application Failures

Network Failures



Application

Service A

Service B

Service 1

Service 2

Service 3

Service 1

Service 2

Service 3

# THANK YOU

JAIPRAKASH PHERWANI

JAIPRAKASH.PHERWANI@GMAIL.COM

+91-9930913355