

Dublin, 8.10.2025



ANDI DEUSCHL

Dynatrace

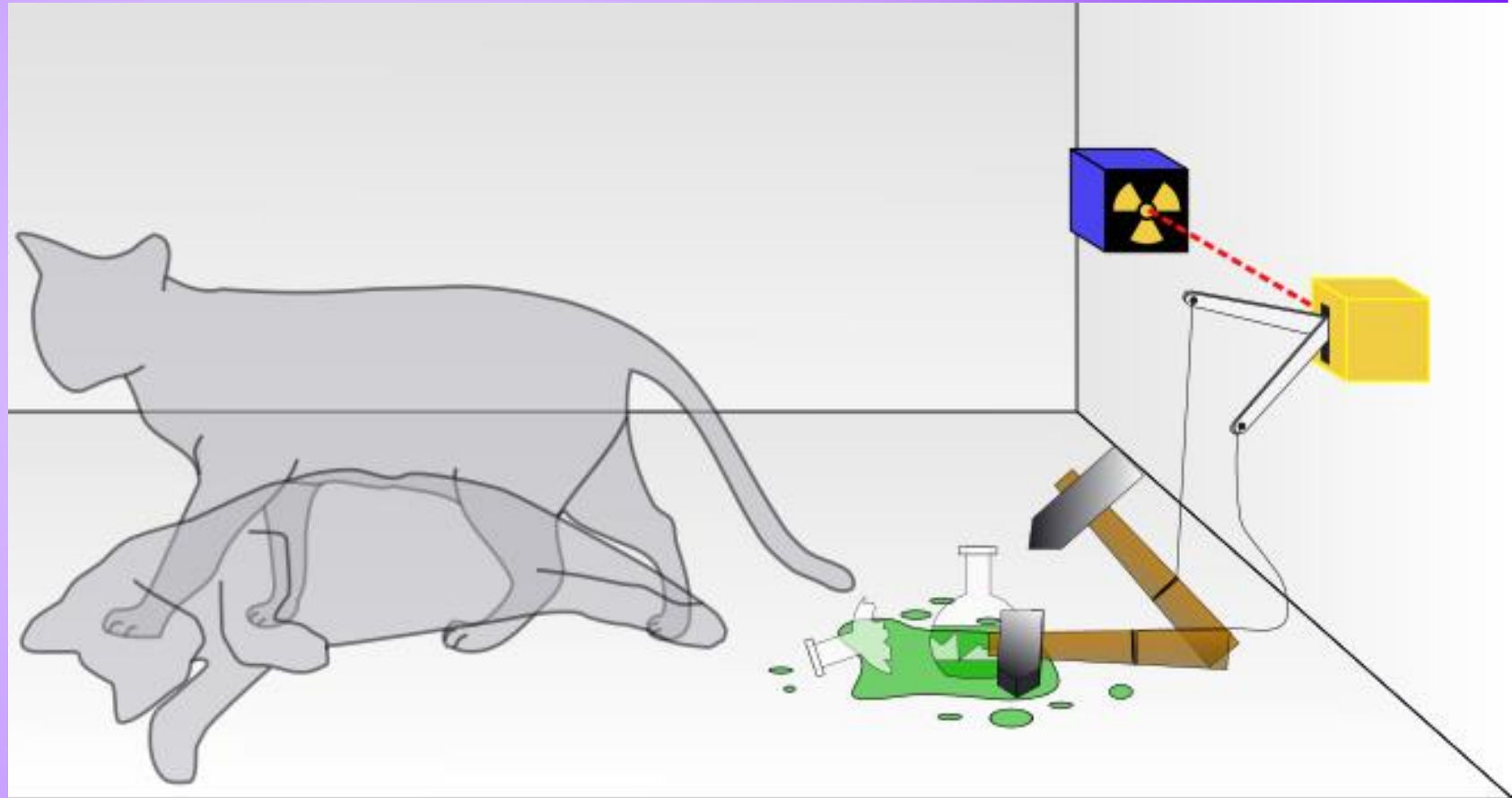
The Un-Incident

EXTRACTING VALUE FROM THE
GRAY AREA OF INCIDENT
RESPONSE

Schrödinger's Outage

Is it an ~~Incident?~~ Is it
not?

How is the ~~cat~~ doing?



This Photo by Unknown Author is licensed under [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/)

About Me

- Product Lead for Delivery, Reliability & Security at Dynatrace
- Former SRE Lead, CTO, CISO, Ops, ..
- Always into Incident Management
- SRE by heart
- Based in Vienna / Austria



The Un-Incident

Not a new concept or methodology, rather a
focus on the **ambiguity**

The Un-Incident

Undeclared
Neglected
Debated

30-60%

of incidents don't make it in formal tracking
is productive

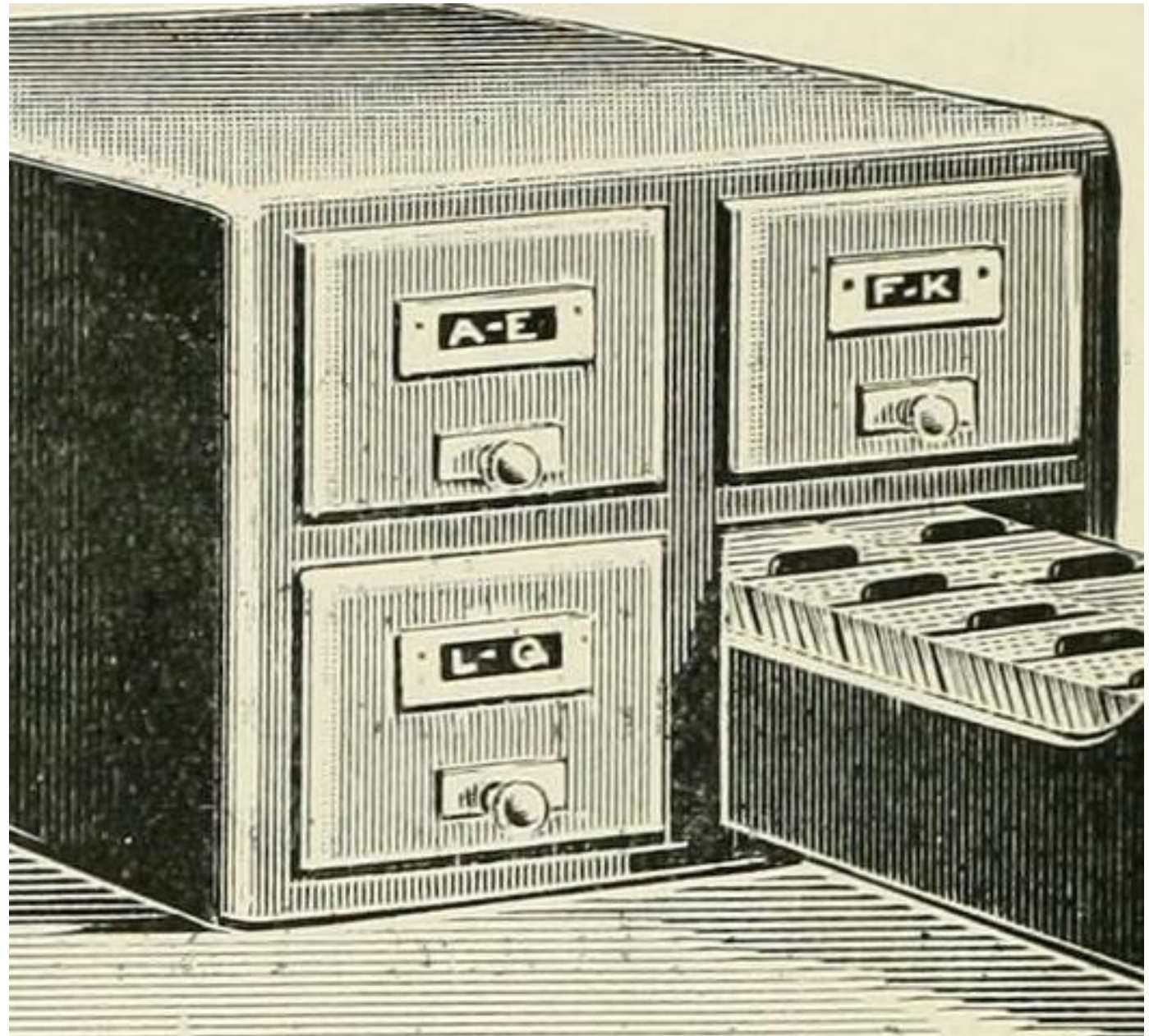
Source: own experience

The Taxonomy of Un-Incidents

No-CIs

Customer-impacting but
non-critical issues

Does not fit well into an
existing classification



[This Photo](#) by Unknown Author is licensed under [CC BY-SA](#)

“NOF” Incidents

“It’s **N**ot **O**ur **F**ault”

External and user driven

e.g. 3rd party, misconfiguration



[This Photo](#) by Unknown Author is licensed under [CC BY-NC](#)

Near Miss

Events that did not escalate due to **luck** or **preventative actions**

Phew moments, instead of **Why?**

Recommended read:

<https://surfingcomplexity.blog/2025/02/01/youre-missing-your-near-misses/>



[This Photo](#) by Unknown Author is licensed under [CC BY-SA](#)

Fear Miss

Escalation by anxiety

Overhead by overreaction

Help button syndrome



[This Photo](#) by Unknown Author is licensed under [CC BY-NC-ND](#)

Fear Miss

Escalation by anxiety

Overhead by overreaction

Help button syndrome

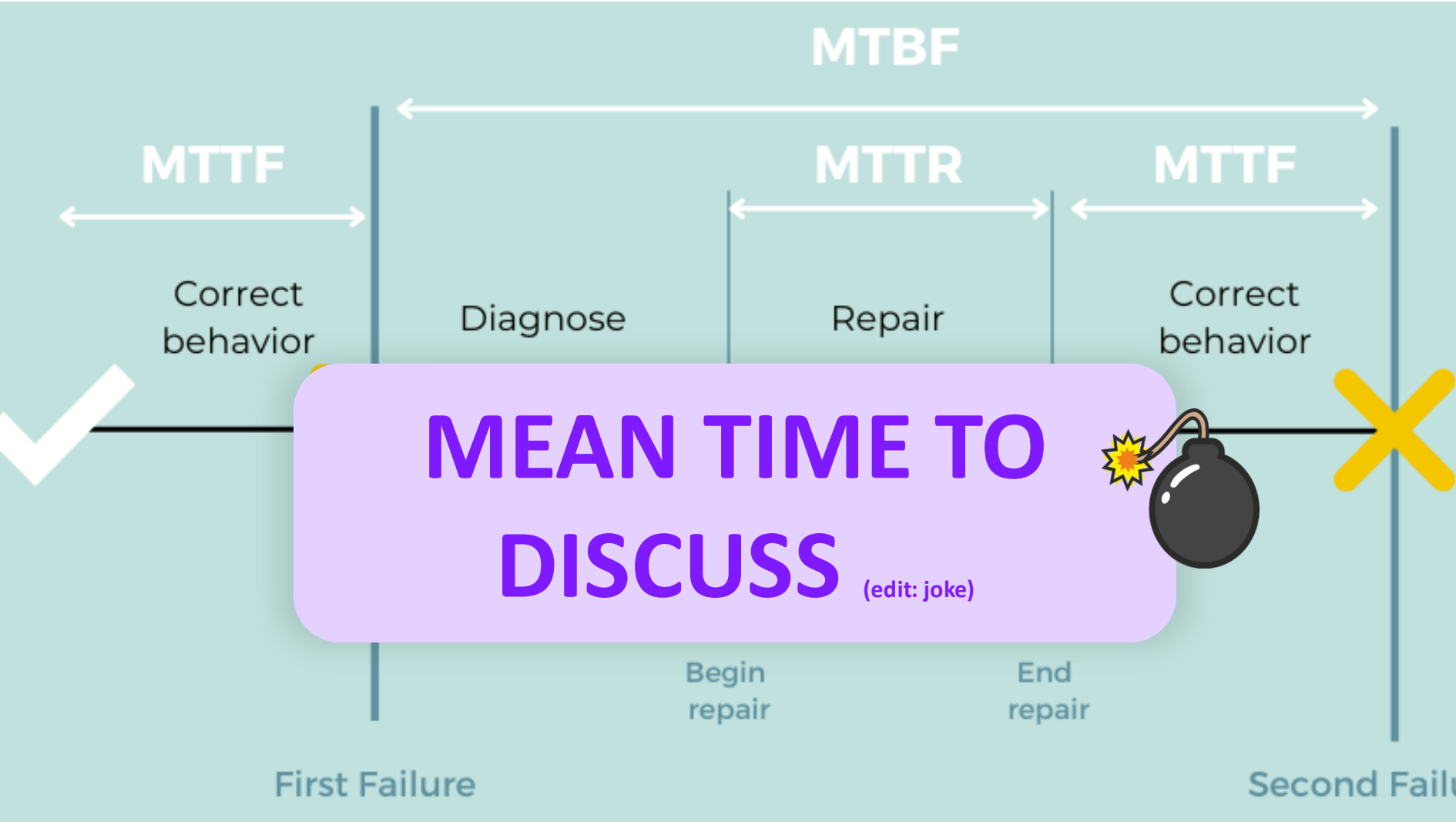


[This Photo](#) by Unknown Author is licensed under [CC BY-SA](#)



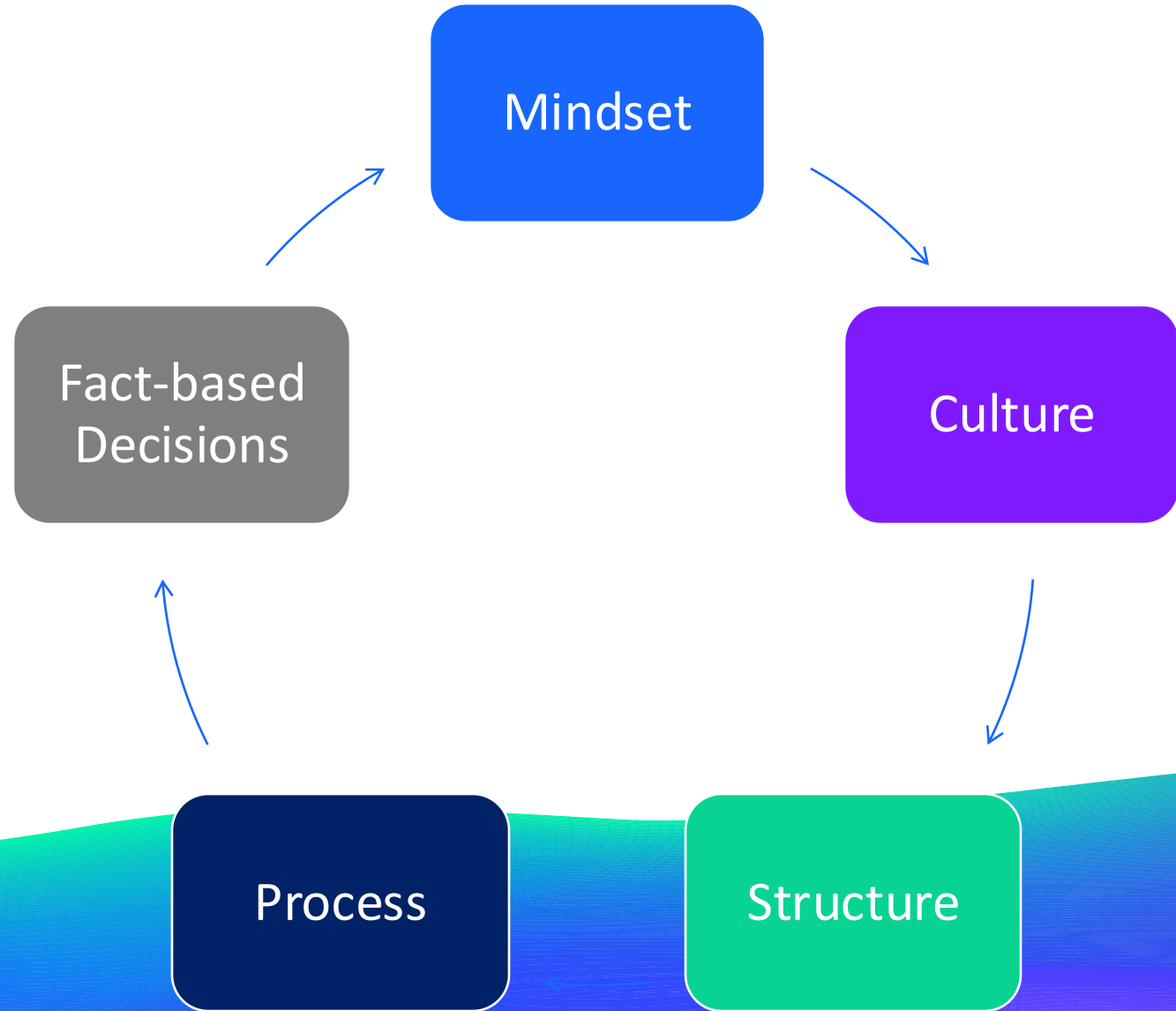
[Photo](#) by Unknown Author is licensed under [CC BY-NC-ND](#)

How to Measure Un-Incidents

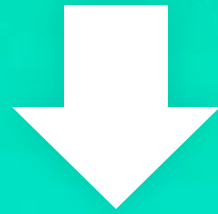


The Gray Zone Playbook

- No-CI
- Near Miss
- It's Not Our Fault
- Fear Miss



SHOULD THIS BE AN INCIDENT?



WHAT CAN WE LEARN?

Psychological Safety around Ambiguity



Don't block people to declare



Appreciate the declaration



Fact-Based Triage

Reduce Reliance on Gut Feeling when declaring Incidents

Build trust in observability

Clear view on the impact of a problem allows fact based decision

Near Miss

Surface gray areas

Detect subtle degradation patterns before they escalate

SLO, SLO, SLO, ..

Near Miss

No-CI

Fear Miss

Track and tag “lucky saves”

Make “Near Misses” visible and reviewable

Near Miss

Self-service AI-supported incident response

Observability data with Context and actionable advice to speed up remediation!

JavaScript error rate increase

Closed P-2508359 Error Started at Aug 6, 2025, 7:57 AM for 36 min

Explain with AI

Affected from...	Affected ser...	Affected inf...	Affected syn...	Affected us...	SLOs	Events
1	6	6	2	1.47k	3	38

Explain what happened in the Davis problem P-2508161, why it happened, and actionable steps to remediate it.

Explanation of Davis Problem P-2508161

- What Happened:**
 - The JavaScript error rate increased significantly, affecting 8 entities overall.
 - The root cause was identified as a failure rate increase in the `otelDemo.PaymentService`, where the error rate rose to 51.25% over the last 30 minutes, deviating from the baseline of 0%.
 - Other affected services include `astroshop-checkout`, `frontend`, and `astroshop.playground-dev.demoability.dynatrace.com`, all showing elevated failure rates across specific endpoints.
- Why It Happened:**
 - The issue correlates with a deployment triggered by ArgoCD Sync for the Astroshop application (commit `934ac0`).
 - The deployment likely introduced changes that caused service implementation issues, resource saturation, or increased load, leading to elevated error rates.
 - Specific endpoints such as `Charge`, `PlaceOrder`, and `/api/checkout` experienced significant failure rate increases.

Actionable Steps to Remediate

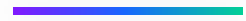
- Rollback Deployment:**
 - Revert the changes introduced by the ArgoCD deployment (commit `934ac0`) to restore the system to its previous stable state.
- Investigate Code Changes:**
 - Review the code changes in the pull request associated with commit `934ac0` to identify potential bugs or misconfigurations.
- Analyze Logs and Metrics:**
 - Use Dynatrace to analyze logs and metrics for the affected services and endpoints to pinpoint the exact cause of the failure (e.g., resource saturation, unhandled exceptions).
- Test and Validate Fixes:**
 - Implement fixes for identified issues and test them thoroughly in a staging environment before redeploying to production.
- Monitor Post-Deployment:**
 - After redeployment, closely monitor the error rates and service health using Dynatrace to ensure the issue is resolved and no new problems arise.
- Optimize Resource Allocation:**
 - If resource saturation is suspected, evaluate and optimize resource allocation for the affected services to handle increased load effectively.

Send a message

AI-generated results might be inaccurate. Learn more about [Davis CoPilot](#).

Structure before Sentiment

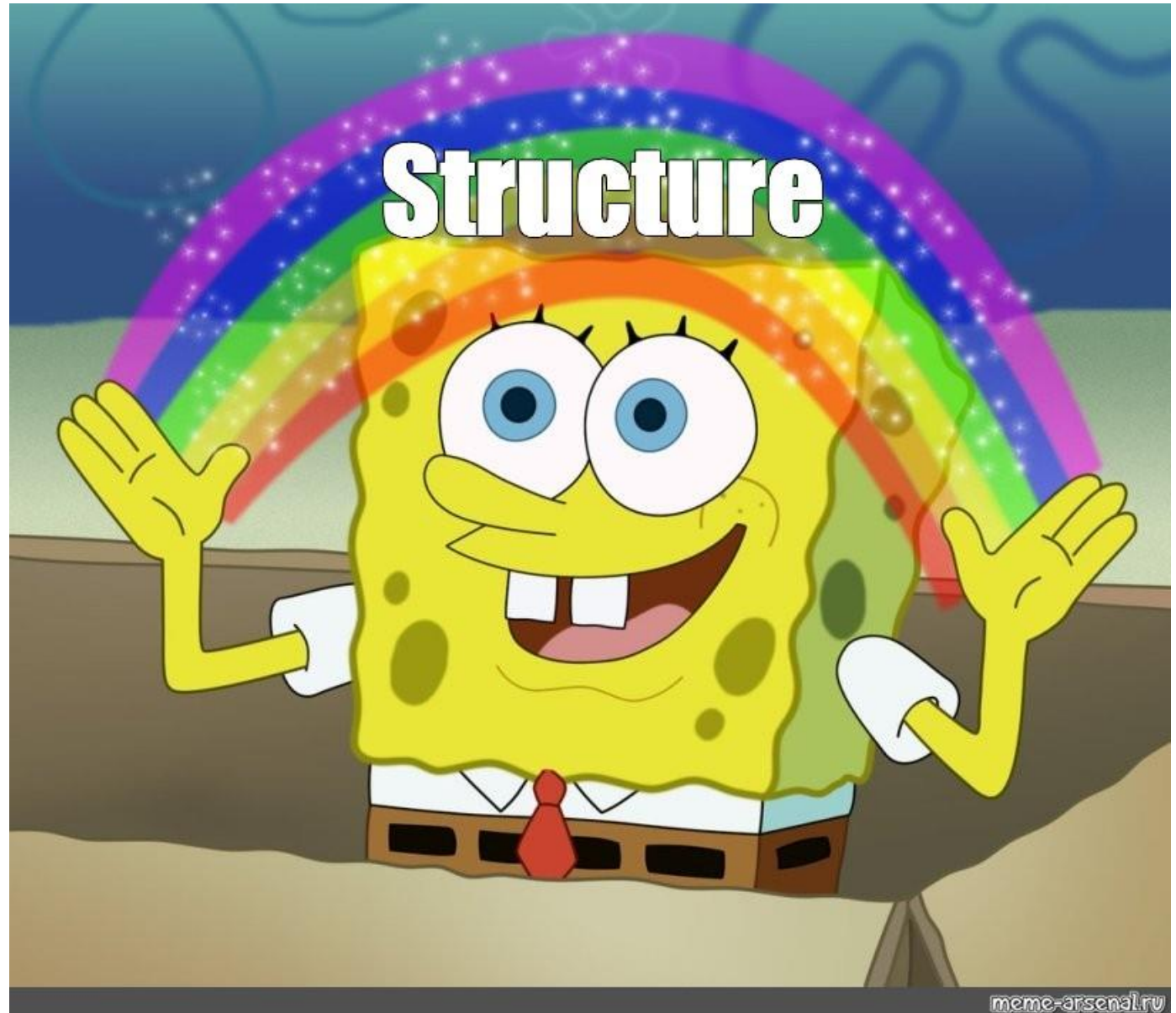
Structure before Sentiment



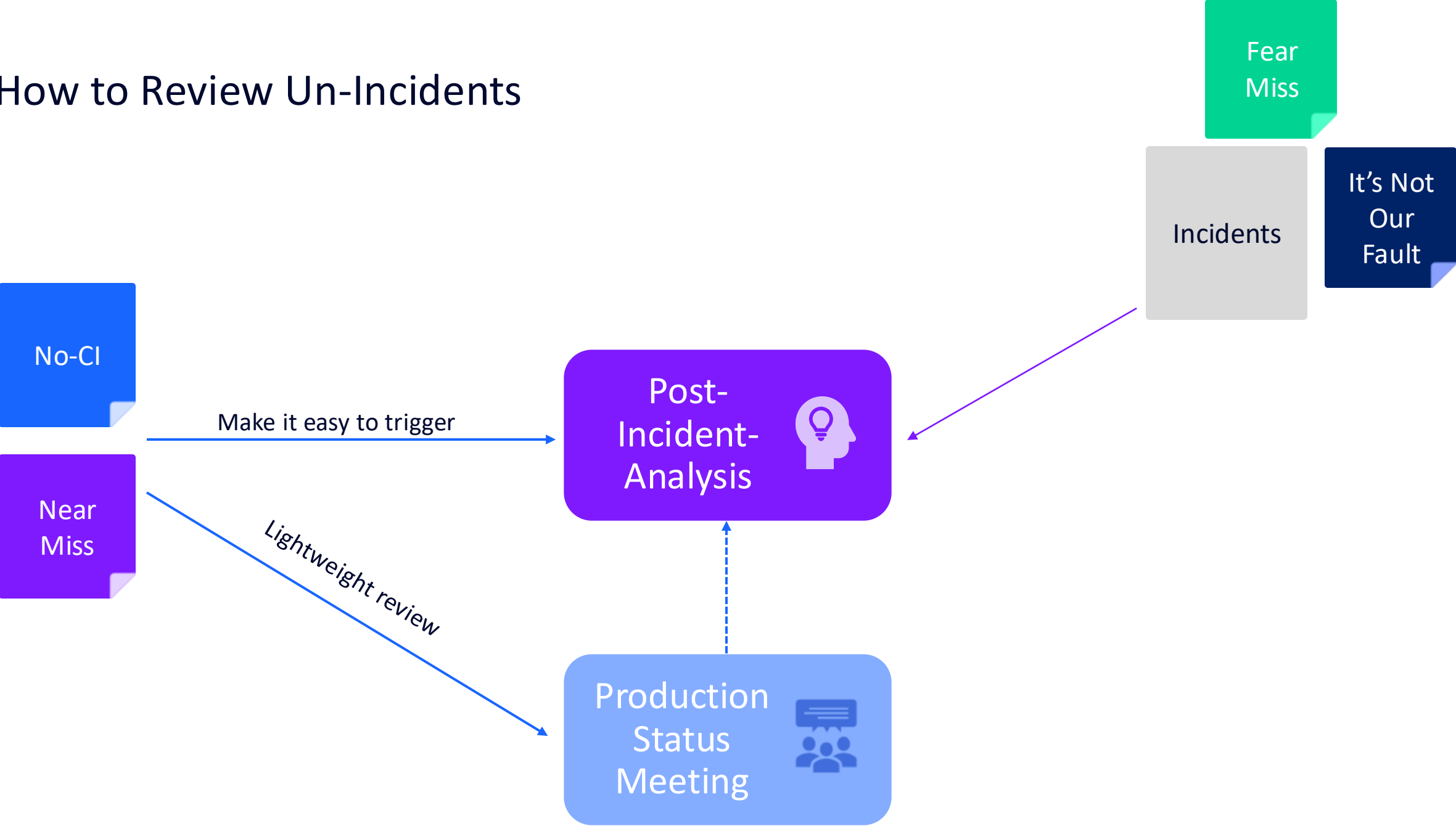
Clarity

Improves Mindset

Confidence



How to Review Un-Incidents



Incident (Process) Reviews

Analyze incident patterns

Review the efficiency of your response process

Might reveal missing alternatives for Un-Incidents

Having conducted an incisive 360 degree review of all incidents in the last month we've determined that "shit happens"



[This Photo](#) by Unknown Author is licensed under [CC BY](#)

Guided Escalation Paths

Fear Miss

It's Not Our Fault



Finding: declared incidents as there was no better choice



Review incident classification



Provide non-incident support channels

- Bug Process
- Request for Assistance
- Clear timelines
- Options to escalate

Escalation Reveals Product Improvements



This Photo by Unknown Author is licensed under [CC BY-NC-ND](https://creativecommons.org/licenses/by-nc-nd/4.0/)



This Photo by Unknown Author is licensed under [CC BY](https://creativecommons.org/licenses/by/4.0/)

Patterns for Product Design based on NOF Incidents

Impact != Fault

Think Like a Customer

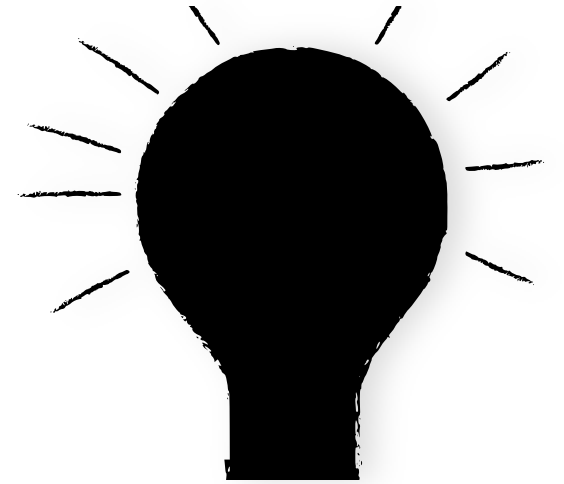
Guardrails Matter

Shared Responsibility

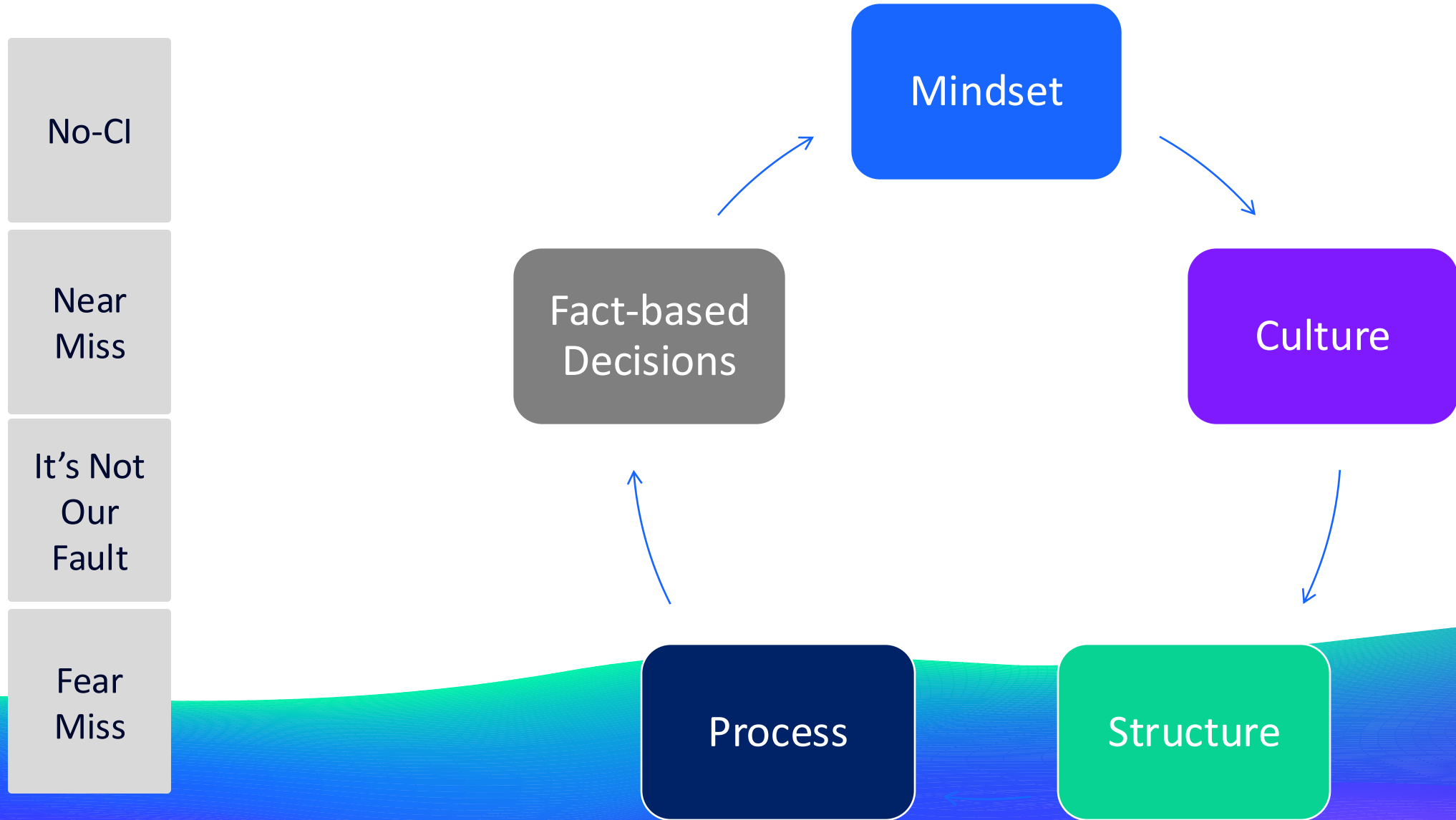
Design for Misuse

Clarity Prevents Escalation

Surface Breaking Changes



Recap: The Gray Zone Playbook



TO **INCIDENT**, OR NOT TO
INCIDENT, THAT IS **NOT**
THE QUESTION



This Photo by Unknown Author is licensed under [CC BY-SA-NC](#)