Why Most SLO Implementations Fail

Amin Astaneh

Certo Modo · certomodo.io

Who's talking

Amin Astaneh - DevOps and SRE consultant, Certo Modo

- **9 years at a rocket ship** - scaled production through hypergrowth, built the company-wide DevOps/SRE program
- **3 years in big tech** - reliability overhaul of an internal platform
- **3 years consulting** - reliability assessments and implementations, startups to Fortune 500

CERTO MODO


FREESPIRIT

Let's Start With A Statistic

In a recent poll of over 100 organizations:

CERTO MODO

“81% of organizations either have no SLOs at all (26%) or have SLOs that exist but don't influence any decisions (58%)- leaving only 16 of 102 orgs using SLOs to actually drive action.”

(reliabilityeconomics.com)

Most organizations aren't benefiting from SLO.

SLOs aren't failing because of bad metrics or tools...

Reliability has no seat at the table.

SLOs are completely useless unless they inform your roadmap.

We'll Cover...

- 4 major antipatterns, personally observed in the field.
- The costs of not addressing them
- The SLO Maturity Model
- 5 Questions to Quickly Assess Your Team

Antipattern 1: Toothless SLOs

They did everything (mostly) right.

- Dedicated SRE team
- SLOs defined per product value stream
- Dashboards, alerting, incident response
- Postmortem process

CERTO MODO

None of this made a material impact on production reliability.

CERTO MODO

Who (outside of SRE) changed their behavior because of an SLO violation?

SRE was trying to improve reliability
all by themselves.

The Toothless SLO

SLOs that generate signal, but carry no organizational authority.

> "We have SLOs." ✓
> "SLO performance gates feature releases." ✗
> "Product changes the roadmap when we spend the error budget." ✗

SRE cannot improve the reliability of a system by themselves.

Reliability is everyone's job.

Antipattern 2: The Out of Date Dashboard

A company adopts an SLO monitoring and governance platform.

CERTO MODO

What we found inside.

- SLOs across every major service ✓
- Centralized set of dashboards ✓
- Automatically collected SLIs ✓

CERTO MODO

Last updated: 18 months ago.

The engineer who originally built them left the company long ago.

Meanwhile:
customers were still complaining.

CERTO MODO

The Out of Date Dashboard

SLOs defined at some point, but now are measuring the wrong things.

SLOs are not 'one-and-done'.

Antipattern 3: SLO Vaporware

Meta has virtually unlimited engineering resources.

World-class infrastructure.

Some of the best SREs on the planet.

I joined a team within Meta's Core Systems.

Two weeks before I started, they had committed to a set of SLOs to a major stakeholder eager to onboard.

What those SLOs actually were:

A post in a Workplace group.

No measurement of any kind.
No dashboards.
No alerting.
No enforcement.

To make matters worse:
the targets were impossible.

I had to tell them
they couldn't keep the promise.

It took two years to implement SLOs that were actually useful.

CERTO MODO

SLO Vaporware

A document lacking measurement, tooling, validation, or process.

SLOs require discussion, validation, supporting metrics, a dashboard, and processes.

A wiki page isn't enough.

Antipattern 4: Perpetual Procrastination

We had a major infrastructure incident.

There were questions about how to improve change management processes across the board.

My response:

"We need SLOs."

The CEO's response:

"We need a Change Advisory Board."

His justification:

"We need to walk before we can run."

It wasn't.

A CAB adds process. SLOs add **feedback**.

CERTO MODO

Perpetual Procrastination

Where there's always a reason to not implement SLOs right now at a team or company.

> "We know we need SLOs, but..."
> "We're not ready yet."
> "We need to walk before we can run."

NOW is always the best time to implement SLOs.

NOW is always the best time...

- You don't have an SRE practice without them
- How else will you know if the product is functioning well from the customer's point of view?

CERTO MODO

The pattern across every story is the same.

SRE defines the SLOs...
SRE sees the problems...
SRE gets paged at 3am...
SRE does the writeups and presents the postmortems...

Nobody else changes their behavior.

The long-term costs:

- SRE and observability spend with no improvement in reliability
- SRE loses credibility with engineering- new processes, same pain
- Customer churn risk grows silently while the dashboard shows green

Who has the authority to stop a release based on reliability data?

What happens to their career if they do?

Where do you sit?

The SLO Maturity Model

https://gitlab.com/certomodo.io/slo-maturity-model

Level	State	Antipattern
0	No commitment	Perpetual Procrastination
1	Not defined yet	SLO Vaporware
2	Defined, not consistently used	Out of Date Dashboard / Toothless SLO
3	Consistently applied in team processes	-
4	Embedded in workflows	-
5	Continuously improved	-

CERTO MODO

All orgs need to be at least here...

Level 3: Defined

SLOs guide incident response and work prioritization. Adoption has spread beyond a single individual.

- SLOs exist
- Alerts fire
- Cross-functional team responds to error budget burn
- SLO violations can delay a release

CERTO MODO

Many teams are actually here.

Level 2: Repeatable

SLOs exist and are visible, but the team's adoption is inconsistent.

- SLO processes are driven entirely by SRE
- Alerts fire, but only SRE responds
- No authority to enforce consequences for violations
- Product has never heard of an error budget

CERTO MODO

How to scale company-wide...

Level 4: Capable

SLOs are embedded in day-to-day engineering and release workflows.

- SLOs gate releases in CI/CD
- Product launches require SLO sign-off
- Cross-functional input: CS, Product, Engineering
- Error budget drives release decisions

CERTO MODO

How to become world-class...

Level 5: Efficient

Continuously improved and aligned to customer expectations. Error budget use is deliberate and strategic.

- Regular structured SLO reviews company-wide
- Error budget spent strategically, not just through incidents
- Teams sometimes induce failure to validate resilience
- SLOs evolve with the product, not behind it

CERTO MODO

The jump from Level 2 to Level 3 isn't technical.

It's the moment reliability gets a seat at the table.

What this practically means

- **Exec commitment with teeth**- we have explicit permission to slow down when needed
- **Error budget policy that's enforced**- violations have documented business rules, applied consistently
- **Product in the room**- SLO design, reviews are not an SRE-only meeting.

Five questions I ask every team:

1. Show me your SLO dashboard.

2. How and when do you use it?

3. Who defined the SLOs, and when were they last changed?

4. What happens during a high burn event?

5. What happens when your error budget is spent?

The answers tell you everything.

If you get silence on questions 4 and 5:
you're probably at Level 2.

CERTO MODO

SLOs are completely useless unless they inform your roadmap.

Do one thing this week.

Include your product manager in your next SLO review.

Thank you.
Questions?

Slides, the maturity model, and how to contact me:

certomodo.io/talks/why-slos-fail