A Dual Approach to Accountability Engineering

Building Healthy SRE Teams and a Culture of Accountability at Reddit
But, first!
Familiar Concepts

Observability
Availability
Production
Infrastructure
Platform

Cloud
Systems
Performance
Scalability
Site Reliability
(Still?) **Familiar Titles**

- **Observability** Engineer
- **Availability** Engineer
- **Production** Engineer
- **Infrastructure** Engineer
- **Platform** Engineer
- **Cloud** Engineer
- **Systems** Engineer
- **Performance** Engineer
- **Scalability** Engineer
- **Site Reliability** Engineer
Familiar Disciplines

- Observability Engineering
- Availability Engineering
- Production Engineering
- Infrastructure Engineering
- Platform Engineering
- Cloud Engineering
- Systems Engineering
- Performance Engineering
- Scalability Engineering
- Site Reliability Engineering
wut would you say you do here?

Umm

Observability Engineering Team
Availability Engineering Team
Production Engineering Team
Infrastructure Engineering Team
Platform Engineering Team

Cloud Engineering Team
Systems Engineering Team
Performance Engineering Team
Scalability Engineering Team
Site Reliability Engineering Team
Definition

Site Reliability Engineering

definition

a set of principles and practices that applies aspects of software engineering to IT infrastructure and operations. SRE claims to create highly reliable and scalable software systems.

https://en.wikipedia.org/wiki/Site_reliability_engineering
LISA made LISA obsolete (that's a compliment!)

Goodbye LISA! Hello SRECon!

October 27, 2022

CULTURE

Authors: Thomas A. Limoncelli

Article shepherded by: Rik Farrow

SRE vs Platform Engineer: Can’t We All Just Get Along?

Is it really SRE vs platform engineer? Or is there a way platforms can take site reliability to the next level?

Aug 30th, 2023

by Jennifer Riggins

Becoming a Platform Engineer

I am a DevOps/SRE and I have interviewed 10s of people to join the platform team of our company (Fortune 500, hundreds of devs across tens of teams). One of the main issues we have noticed among candidates is the lack of basic understanding of how everything fits together and how to view internal tooling.

Becoming a decent DevOps and later platform engineer has been quite the journey for me, here's my 2c on what I think DevOps/SREs should focus on when looking at internal tooling and related roles.
YOU’RE A DEVOPS ML INFRASTRUCTURE PLATFORM SYSTEMS SCALABILITY PRODUCTION SITE RELIABILITY ENGINEER?

TELL ME MORE!
LISA made LISA go a complete circle.

Goodbye LISA!

Becoming a Platform Engineer

I am a DevOps/SRE and I have interviewed (Fortune 500, hundreds of devs across the US) candidates is the lack of basic understanding of tooling.

Becoming a decent DevOps and later platform engineer has been quite the journey. This is my 2c on what I think DevOps/SREs should focus on when looking at internal tooling and related roles.
Stages of Group Development

Forming
Storming
Norming
Performing

How do you keep your team here as long as possible?
Me: Determined to eliminate ambiguity and build healthy teams.

Me to me: Just rename and reorg.
Stages of Group Development

- Forming
- Storming
- Norming
- Performing
- Adjourning

How do you avoid ending up here?
What keeps an SRE Team **healthy**?

- Focus
- Belonging
- Purpose
- Quality of Outcomes

- Shared Vision
- Recognition
- Agency
- Mastery
- Creativity
I’m Anthony, I lead teams at Reddit and we call ourselves Site Reliability Engineering!
What’s **in scope** for a Reddit SRE Team?

Let’s do it!

- **Observability**
- **Cloud**
- **Availability**
- **Systems**
- **Production**
- **Performance**
- **Infrastructure**
- **Scalability**
- **Platform**
- **Reliability**
Progression with Professional Development

Value

Junior

Senior

Staff+

career

concepts

team / organizational boundaries

experience
Project Based Groupings
Building Healthy SRE Teams with Accountability Engineering
DISCLAIMER

I am not advocating for the adoption of Accountability Engineering as a formal concept.

You should not attempt to hire an Accountability Engineer, they don’t exist.

If at the end of this talk you’ve decided to form an Accountability Engineering Team, find me.
SRE Accountability Engineering Principle #1

SRE Teams are held accountable for an engineering deliverable.
What keeps an SRE Team **healthy**?

- Focus
- Belonging
- Purpose
- Quality of Outcomes

- Shared Vision
- Recognition
- Agency
- Mastery
- Creativity

I’m so happy!
SRE Accountability Engineering Principle #2

SRE Teams build solutions that provide an engineering organization with the data and incentives they need to hold itself accountable to agreed upon standards and goals.
What's in scope for any Engineering Team?

- Observability
- Availability
- Production
- Infrastructure
- Platform

Cloud Systems
Performance
Scalability
Reliability

srsly? yep.
What keeps an SRE Team healthy?

- Focus
- Belonging
- Purpose
- Quality of Outcomes

- Shared Vision
- Recognition
- Agency
- Mastery
- Creativity

I’m so happy!
5 Steps
For Implementing Accountability Engineering Principles
Step 1 Plan with your partner teams

Prioritization & Cross-Functional Alignment
Step 2  Commit to an engineering deliverable

Prioritization & Cross-Functional Alignment

High recognition  High control

Inaction
Lack of ownership
Poor outcomes
Low morale

Low recognition  Low control
Step 2  Commit to an engineering deliverable  ⚠
Step 3 Select Projects that Satisfy Your Team

**Direct Accountability**
- Agency to solve problems
- Technically challenging
- High reward and satisfaction

**Inaction**
- Lack of ownership
- Poor outcomes
- Low morale

**Prioritization** & Cross-Functional Alignment

Goal: near-term engineering improvement
Step 4 Select Projects that Drive Cultural Improvement

- Prioritization & Cross-Functional Alignment
- Inaction
  - Lack of ownership
  - Poor outcomes
  - Low morale
- Diffuse Accountability
  - Empower local improvement
  - Sociotechnically challenging
  - Lacks guarantees

Goal: long-term cultural improvement
Step 5
Build a Roadmap and Manage Interrupts

Set your team up for success.
**Prioritization & Cross-Functional Alignment**

**Direct Accountability**
- Agency to solve problems
- Technically challenging
- High reward and satisfaction

**Inaction**
- Lack of ownership
- Poor outcomes
- Low morale

**Diffuse Accountability**
- Empower local improvement
- Sociotechnically challenging
- Lacks guarantees

**Goal:**
- near-term engineering improvement
- long-term cultural improvement
What keeps an SRE Team healthy?

- Focus
- Belonging
- Purpose
- Quality of Outcomes
- Shared Vision
- Recognition
- Agency
- Mastery
- Creativity

I'm so happy!
**Prioritization & Cross-Functional Alignment**

**Direct Accountability**
- Agency to solve problems
- Technically challenging
- High reward and satisfaction

**Inaction**
- Lack of ownership
- Poor outcomes
- Low morale

**Diffuse Accountability**
- Empower local improvement
- Sociotechnically challenging
- Lacks guarantees

**Goal:**
- near-term engineering improvement
- long-term cultural improvement
What keeps an SRE Team healthy?

Focus
Belonging
Purpose
Quality of Outcomes

Shared Vision
Recognition
Agency
Mastery
Creativity

I’m so happy!
Focus Block: Culture and Diffuse Accountability
SRE Accountability Engineering Principle #2

SRE Teams build solutions that provide an engineering organization with the data and incentives they need to hold itself accountable to agreed upon standards and goals.
Step 4  Select Projects that **Drive Cultural Improvement**

- **Prioritization & Cross-Functional Alignment**

  - Inaction
    - Lack of ownership
    - Poor outcomes
    - Low morale

  - Diffuse Accountability
    - Empower local improvement
    - Sociotechnically challenging
    - Lacks guarantees

- Goal: long-term cultural improvement
What has Reddit done to create a culture of Accountability Engineering?
We built an incident response framework.
We support a service catalogue.
We built a pipeline for publishing and alerting on SLOs.

<table>
<thead>
<tr>
<th>OWNER</th>
<th>REDACTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>SERVICE TIER</td>
<td>0</td>
</tr>
<tr>
<td>TARGET</td>
<td>98.5%</td>
</tr>
<tr>
<td>TYPE</td>
<td>latency</td>
</tr>
<tr>
<td>ROLLING WINDOW</td>
<td>28d</td>
</tr>
<tr>
<td>SERVICE</td>
<td>REDACTED</td>
</tr>
<tr>
<td>TAGS</td>
<td>No Tags</td>
</tr>
</tbody>
</table>
We host a recurring VP-level operational review.

<table>
<thead>
<tr>
<th>ID</th>
<th>Report</th>
<th>Sev (max)</th>
<th>Postmortem</th>
<th>Lead</th>
</tr>
</thead>
<tbody>
<tr>
<td>3529</td>
<td>UI is intermittently not loading</td>
<td>SEV1</td>
<td>no decision</td>
<td>Anthony Sandoval</td>
</tr>
<tr>
<td>3527</td>
<td>Increased error rate</td>
<td>SEV1</td>
<td>no decision</td>
<td>John Looney</td>
</tr>
<tr>
<td>3506</td>
<td>The foo service is degraded</td>
<td>SEV1</td>
<td>Yes Postmortem (due 2023/10/20)</td>
<td>Happy Snoo</td>
</tr>
<tr>
<td>3401</td>
<td>5xxs elevated, 2xxs down n%</td>
<td>SEV1</td>
<td>Yes Postmortem (due 2023/10/04)</td>
<td>Grump Snoo</td>
</tr>
<tr>
<td>2542</td>
<td>Service foo down</td>
<td>SEV1</td>
<td>Postmortem complete</td>
<td>SRE Snoo</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Indicator</th>
<th>28-day window</th>
<th>Delta</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>reddit-service-foo</td>
<td></td>
<td></td>
<td>Tier 0</td>
</tr>
<tr>
<td>foo-endpoint1-availability-envoy</td>
<td>99.92%</td>
<td>-0.03</td>
<td>99.95%</td>
</tr>
<tr>
<td>foo-latency-envoy</td>
<td>150ms</td>
<td>+50 ms</td>
<td>200ms</td>
</tr>
<tr>
<td>reddit-service-bar</td>
<td></td>
<td></td>
<td>Tier 1</td>
</tr>
<tr>
<td>bar-availability</td>
<td>99.85%</td>
<td>-0.03</td>
<td>99.88%</td>
</tr>
<tr>
<td>bar-endpoint1-availability</td>
<td>99.95%</td>
<td>0.00</td>
<td>99.95%</td>
</tr>
<tr>
<td>bar-p99-latency</td>
<td>347ms</td>
<td>+22ms</td>
<td>325ms</td>
</tr>
</tbody>
</table>
We're rolling out a scorecard system
**Prioritization & Cross-Functional Alignment**

**Direct Accountability**
- Agency to solve problems
- Technically challenging
- High reward and satisfaction

**Inaction**
- Lack of ownership
- Poor outcomes
- Low morale

**Diffuse Accountability**
- Empower local improvement
- Sociotechnically challenging
- Lacks guarantees

**Goal:**
- Near-term engineering improvement
- Long-term cultural improvement
Takeaways

It might be painful, but the switch is worth it.

- Don’t model, focus on projects and goals.
- Forge team identity around shared work. Avoid silos.
- Plan cross-functionally, iterate and collaborate.
- Redistribute SRE work fairly. Build a culture of reliability.
THANK YOU!
QUESTIONS?

We’re hiring in Dublin!
https://www.linkedin.com/jobs/view/3732062467