



usenix
**SRE
CON**

SREcon22 Asia/Pacific

Deploying humans at the edge of SRE

Jan Peuker

Integration Engineer, Stripe Singapore (this is *not* a Stripe talk)

December 08, 2022

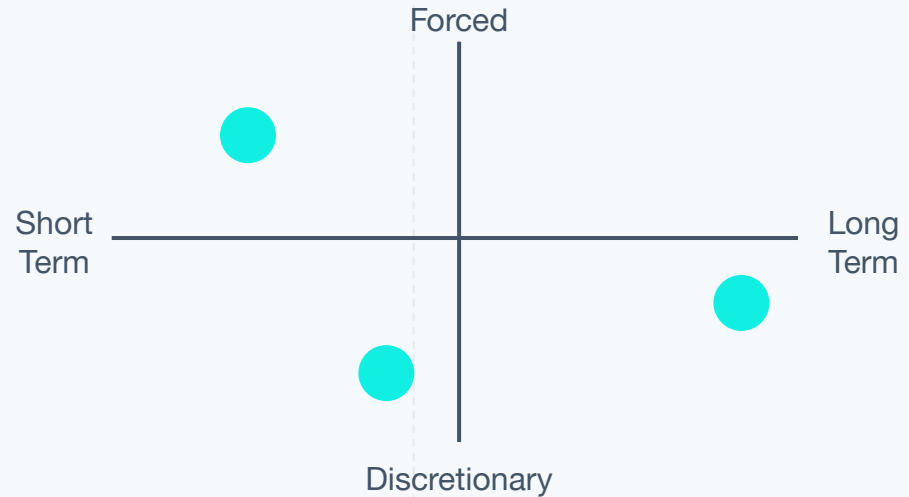
janpeuker@

@janpeuker@infosec.exchange

Innovation:
Listen to your users.
Listen to your users.
Listen to your users.
— Will Larson

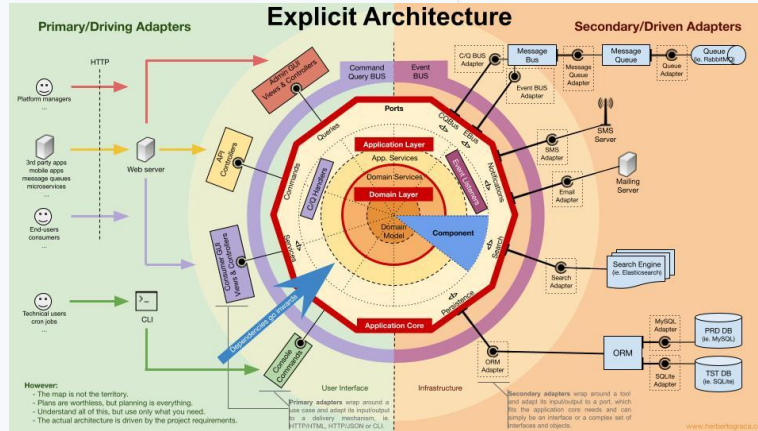
Our Monitoring
doesn't decide
our reliability -
our users do
— Dave Rensin

Before You Deploy
Code, Deploy
Empathy
— Michele Hansen

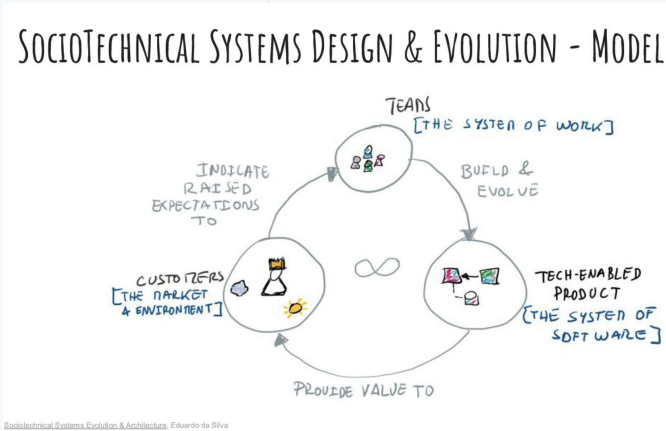


Reliability is perception and SLOs approximate user trust

See Will Larson, How to invest in technical infrastructure; Michele Hansen, Deploy Empathy; Kelsey Hightower; Niall Murphy “Reliability is not the primary goal, socially constructed ‘reliability is the most important feature’ on Resilience Engineering, Safety 1 vs Safety 2; Dave Rensin at SRECon 2017 and SRE Workbook



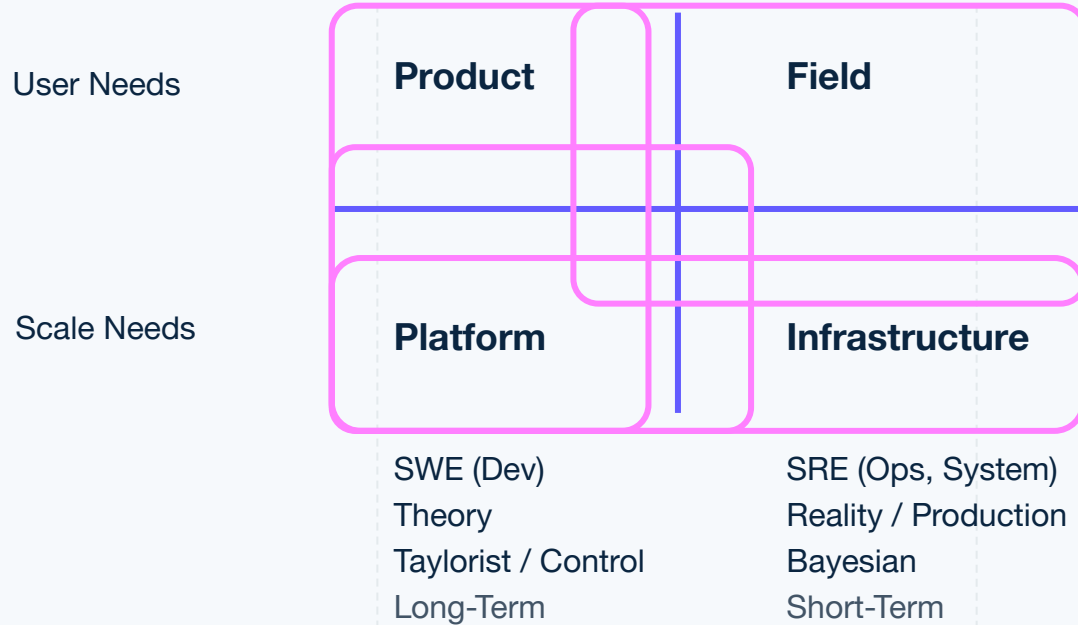
Herberto Graca



Eduardo da Silva

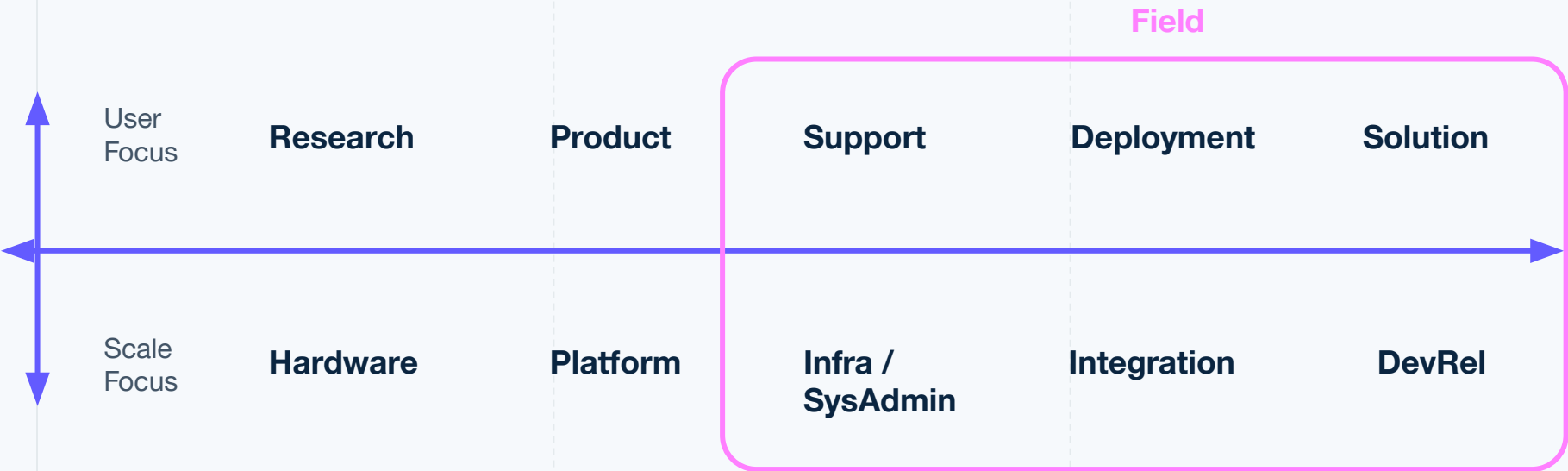
Sociotechnical systems don't "encapsulate" a model

Source Chris W. Clegg 1979 Sociotechnical Theory; see Ruth Malan and Donella Meadows Systems Understanding; Skelton & da Silva GitHub; Alistair Cockburn "Ports and Adapters"; Herberto Graca "DDD, Hexagonal, Onion, Clean, CQRS"; Alvaro Videla's references e.g. Peter Naur, Programming as Theory Building



Engineering practices often omit “field” reliability

Manuel Pais & Matthew Skelton 2019 Team Topologies; Trond Hjorteland / Calvin Pava: Sociotechnical Systems Design for the “Digital Coal Mines”; Charity Majors The Future of Ops Is Platform Engineering; Chris Richardson in CaSE Podcast on Service Chassis; Laura Maguire, Jeli / Howie Guide

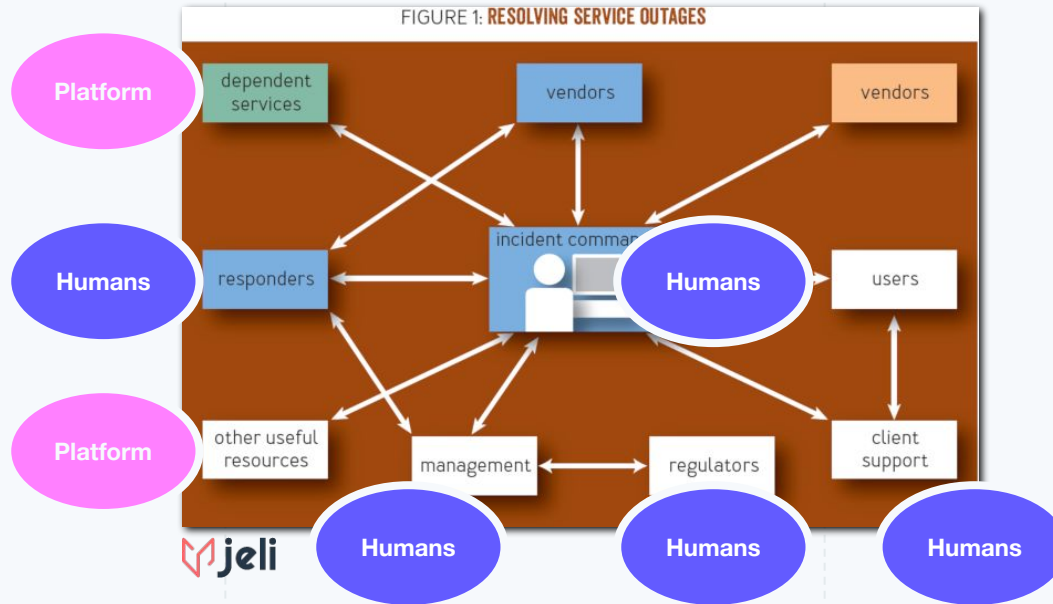


SRE are “user empathy glue” not just platform work

Jessica Kerr, When costs are nonlinear, keep it small; How Buildings Learn, Stewart Brand; Paul Osman, Sociotechnical Lenses into Software Systems; Elizabeth Ayer, Organizational boundary problems; Marisa Leavitt Cohn, Software as a Timely Object; Tanya Reilly “Being Glue”

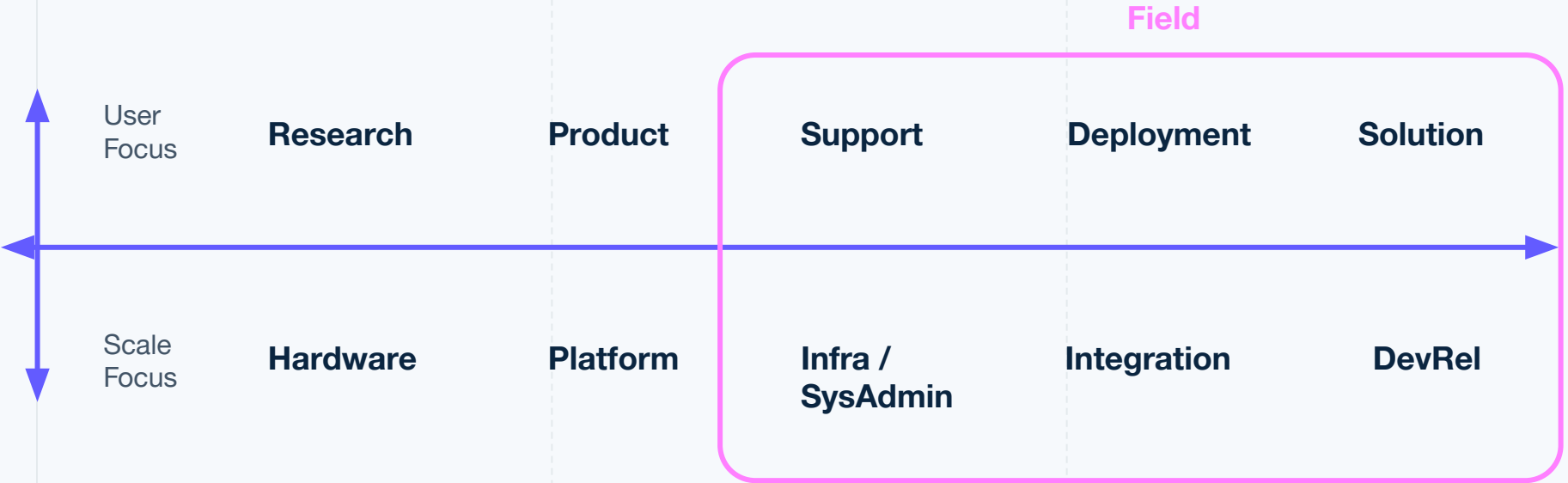
Example 1: User happiness SLOs golden signals

- Talks by Andrew Cowie, Brian Hansen and Yan Yan - Revenue, Margin, ROI
- Netflix season-adjusted stream-starts-per-second (SPS) “*all incident analysis work focuses on the sociotechnical aspects of an incident*” (Hank Jacobs)
- Non-incident Support Cases (e.g. workarounds, near misses (review), retry spikes, degradation)
- DevRel or Account Team / TAM / CSM effort, CSAT / NPS, severity / SLI perception dissonance
- Documentation References from Google Search / Stack Overflow
- API integration shapes (personas) and low precision API usage, source patterns, Heisenbugs
- DORA / Meta-Metrics e.g. new alerts, non-fix case escalations, MTTx (Štěpán Davidovič)
- Zero-SLOs
 - New Users using an old API version (version skew) or pattern (low-margin products)
 - Users flattening out, dropping off or underperforming ARR
 - Incidents or Cases during known events



Example 2: Deploying humans in reliability blind spots

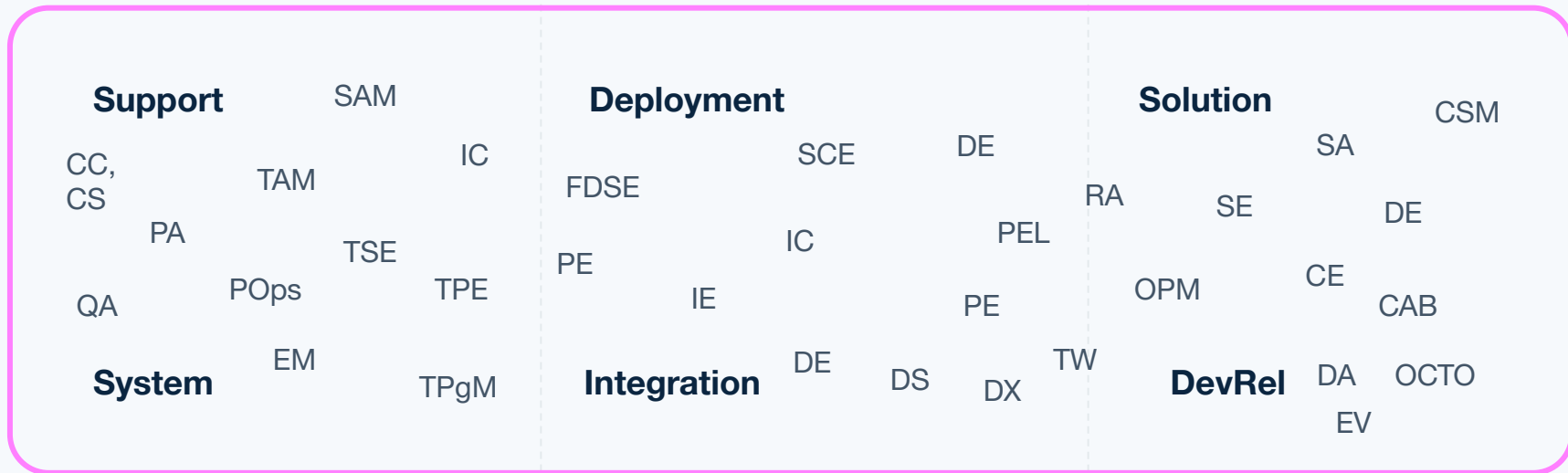
Source Maguire, Managing the Hidden Costs of Coordination; Forsgren & Humble Accelerate “generative culture of the Westrum typology”; Richard I. Cook, Above the Line, Below the Line; Laura Maguire, Jeli / Howie Guide, Fred Hebert “compounding ‘lenses’”; Cindy Sridharan, Know how your org works; Apenwarr Systems design explains the world; Mark Burgess USENIX/LISA Knowledge Management Workshop;



Tech-adjacent roles feed into SRE practices and teams

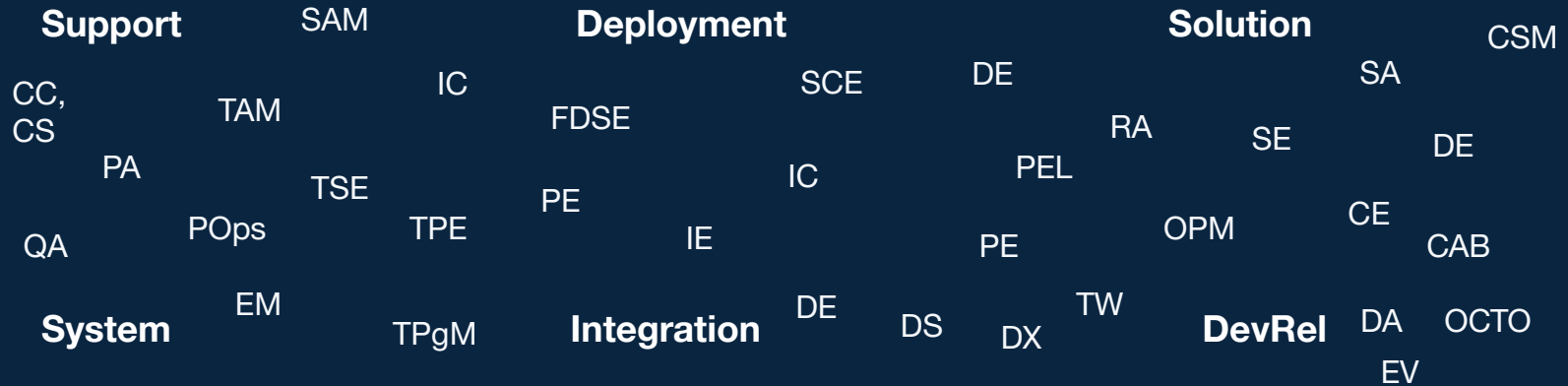
Dave Farley “Foundation of Engineering Discipline is Managing Complexity” and “Feedback in Organization”, “Incremental Design”; Nadia Eghbal, “Code is not a product [but] a living form of knowledge” (Referencing Neal Stephenson on Unix as “Hacker Subculture” rather than product); Cynefin's ordered vs non-ordered

Field



Tech-adjacent roles feed into SRE practices and teams

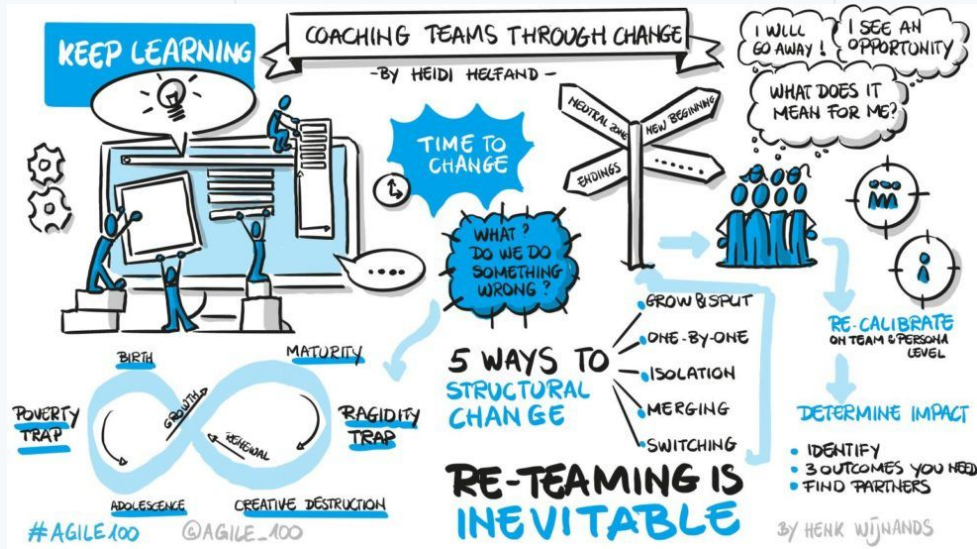
See Emily Webber, Tanya Reilly, Ruth Malan, Kelsey Hightower, Gergely Orosz, Diana Montalion; levels.fyi;
See list of roles at github.com/janpeuker/awesome-tech-roles



SRE practices can be the water The Field swims in

See Emily Webber, Tanya Reilly, Ruth Malan, Kelsey Hightower, Gergely Orosz, Diana Montalion; levels.fyi;

See list of roles at github.com/janpeuker/awesome-tech-roles



Heidi Helfand

ChatGPT can code, we need rotations and residency

Heidi Helfand, Dynamic Reteaming; Catherine Hicks, "It's Like Coding in the Dark"; Felienne Hermans Programmer's Brain, Onboarding; Google Cloud Technical Residency (CTR) and SRE Mission Control see e.g. SRE Workbook chapter "SRE Team Lifecycles". Dave Farley model improvement; Jez Humble, CA(L)MS

The "#DevOps is Dead" content marketing strategy is an exploitative attack on over decade of progress on inclusivity, cultures of trust and safety

— Jesse Robbins

Gulf between approaches that assume a machine is needed vs ones that assume nurturing of an ecosystem is needed.

— Matthew Skelton

SRE is a culture of safety, user empathy and change embedded in engineering practices and principles

See also “Shared responsibilities and shared fate on Google Cloud”; SRE Workbook change management “Nearly all of our problems caused by changes to systems (human or automated)”; Dave Rensin’s SRECon 2018 Talk “SRE is a culture as much as it is a set of principles and practices”



usenix
**SRE
CON**

SREcon22 Asia/Pacific

Thank you and Q&A

#22apac-day2-track1

Jan Peuker

janpeuker@

@janpeuker@infosec.exchange

Please contribute to

github.com/janpeuker/awesome-tech-roles