Fragere 2.0 Chaos Engineering in a Digital Platform

Cloud AppMod Engineer
Chaos Engineering & SRE Advocate
yurynino@google.com

September, 2026



Agenda

01. A Reminder about Fragere

02. The Core Platform Modules

03. Fragere in Action

04. Results and Learnings

05. Conclusions and Future Work

1

A reminder about Fragere

“

Fragere v.1.0 combines CE random failure injections with a designed method using STPA outputs and identifying high-risk control loops as primary.

– *Yury Niño*

NETFLIX

> Chaos Engineering is the discipline of experimenting on a system in order to build confidence in the system's capability to withstand turbulent conditions in production.



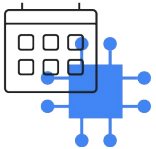
Build **Hypothesis** Around Steady State



Vary **Real-world Events** Changing Variables



Run **Experiments** in Production Environments



Automate Experiments to Run Continuously

https://principlesofchaos.org/

Google

“ DiRT or Disaster Recovery Testing is a coordinated set of events in which a group of engineers plan and execute real and fictitious outages to test the effective response of the involved teams.



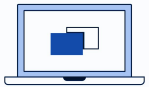
Software

Modifying live service configurations, or bringing up services with known bugs.



Infrastructure

Stress testing large complex architectures, validating SLOs, and ensuring reliability is maintained during disruption.



Access Controls

Including security, compliance, and privacy.



People and Workflows

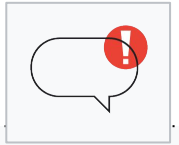
Removing people who might have knowledge or experience.

Fragere: a framework CE + STPA



Identify Hazards

Define unacceptable losses and the system-level hazards using STPA.



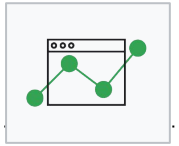
Model Control Structure

Design a model that represents the flow of control and feedback within a system



Identify the UCAs Formulate Hypothesis

Create CE hypothesis for every critical UCA, such as a "commands too late".



Execute Experiments

Run structured CE experiments controlled pre-production environment.

System Theoretic Process Analysis

Chaos Engineering

Fragere v.2.0 is the evolution of this framework in a digital platform that automates the steps of STPA combined with the Chaos Engineering discipline. Fragere v.2.0 provides teams with a unified workspace to visually register hazards, model control structures, configure failure scenarios, and track live chaos experiments.

– *Yury Niño*

2

Core Platform

Modules

Core Platform Modules

Loose Coupling

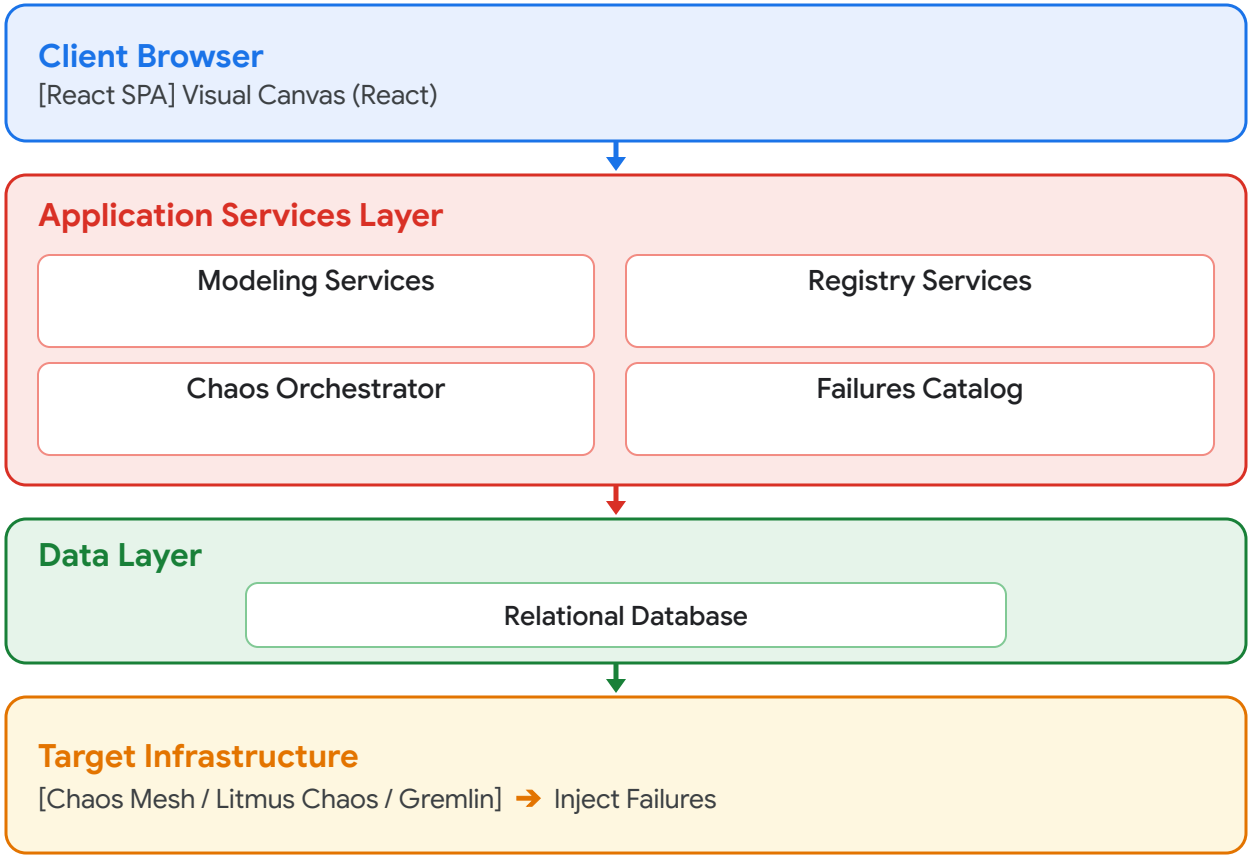
From execution runners to web responsiveness.

Automated Sync

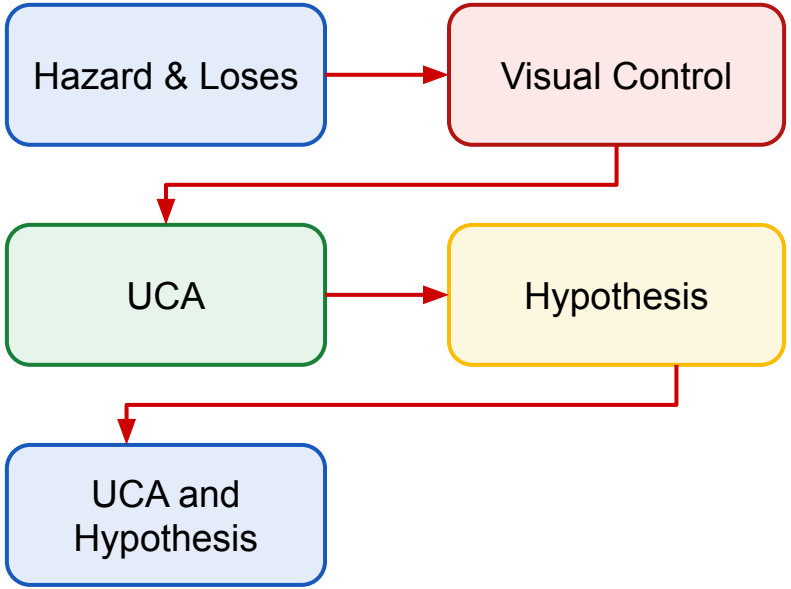
Map target topologies against STAMP control models.

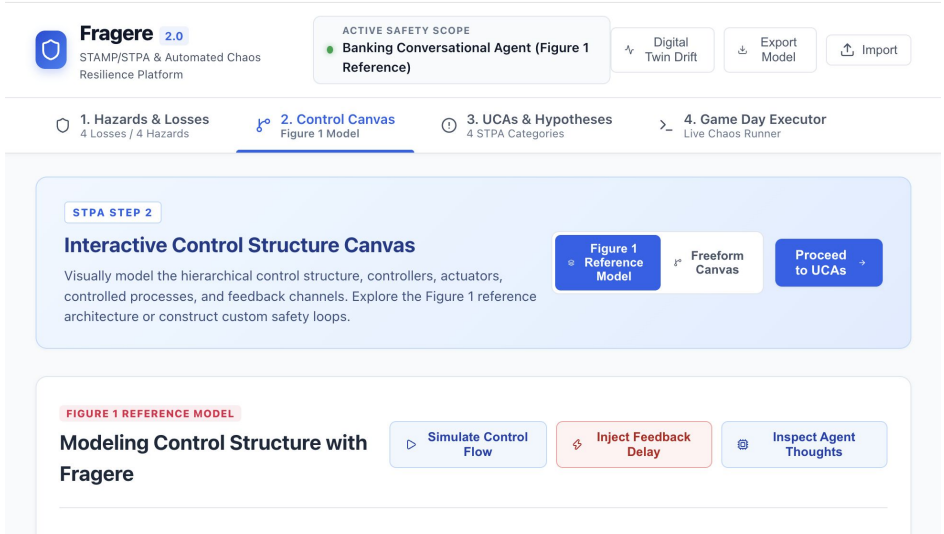
Safety Rollback

Daemons halt experiments if metrics breach thresholds.



Frontend Components





Backend Components

Modeling & Topology Service

Safety & Risk Registry

Chaos Orchestration Service

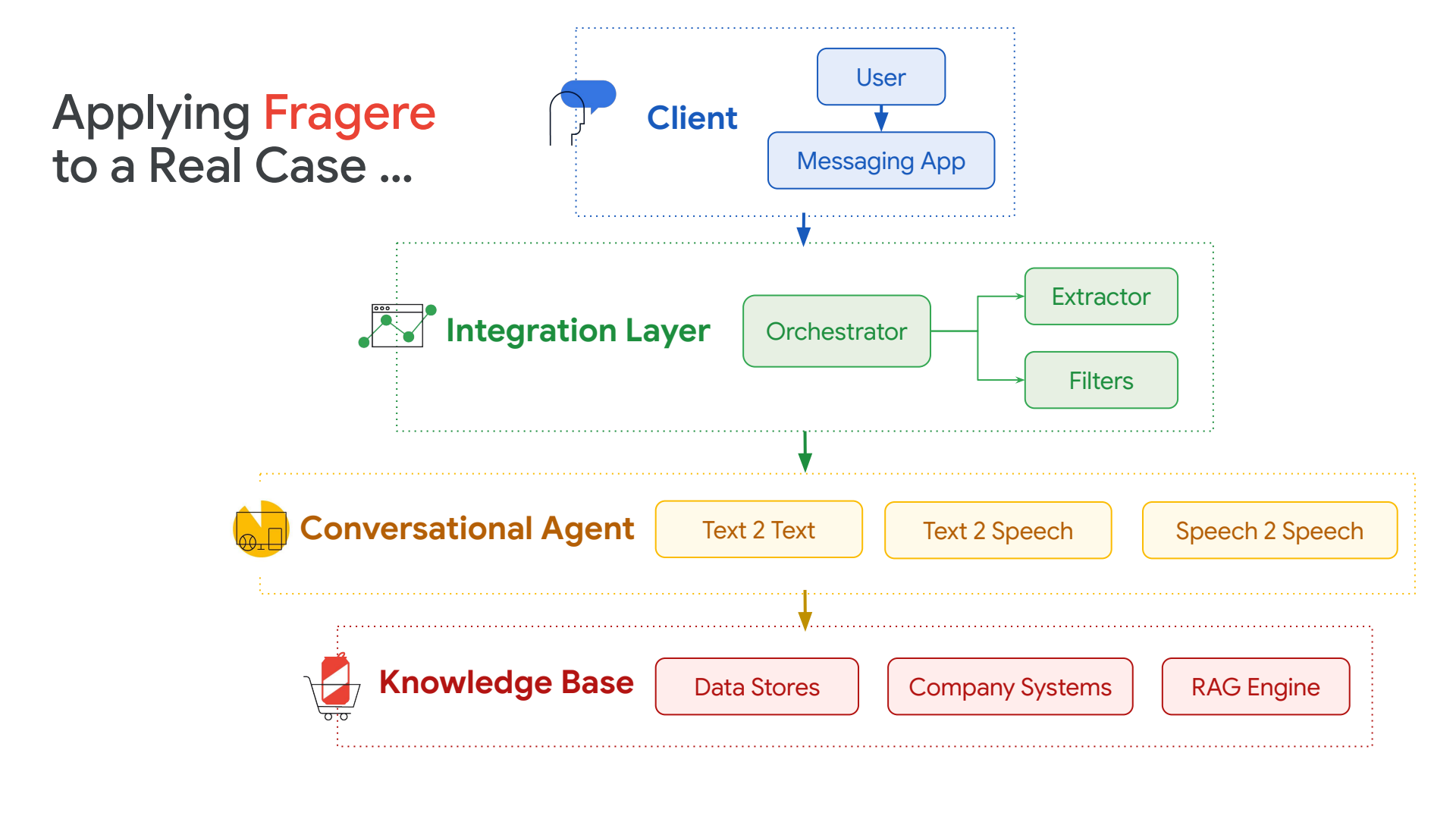
Drift & Sync Worker

3

Fragere

In Action

Applying Fragere to a Real Case ...




Fingers
Control Action
Feedback
Controlled Process

4 Results & Learnings

Resources

6. Best Practices for Implementation

01

Blast Radius Control

Chaos scenario setups require a maximum blast radius thresholds (e.g., maximum duration of 60 seconds or target namespaces).

02

Automated Safety Interlock

If the target application's vital health indicators drop below safety thresholds, Fragere must revert all injected faults.

03

Drift Notification Alerts

Treat structural drifts as urgent warnings on the dashboard, preventing new experiments from launching until the model is updated or resolved.



Youtube

Our takeaways!

01

System Complexity & Formal Safety Models

As system architectures become increasingly complex, traditional component-level reliability is no longer sufficient. To effectively protect digital platforms and financial assets, random failure injections must be guided by formal safety models like STPA to uncover unknown interaction risks.

02

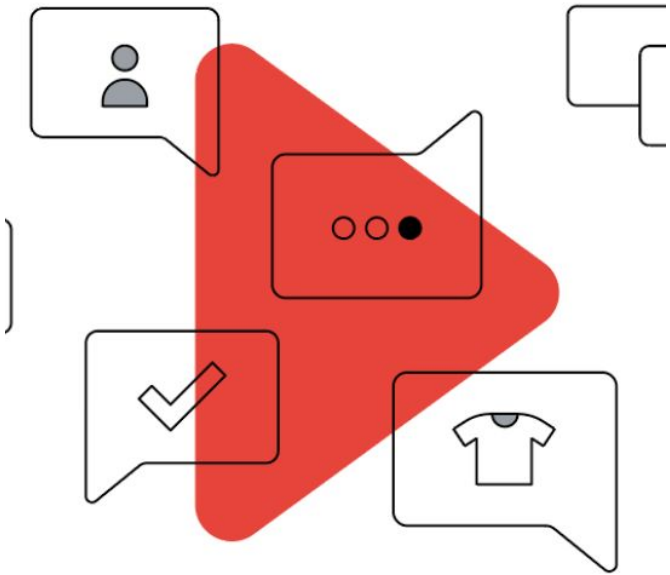
Designing for Safety & Operational Loss Prevention

The goal of designing for safety is to build systems free of issues leading to unacceptable losses or outages. While failure injection is valuable, integrating structured safety models captures critical risks arising from intricate component interactions that traditional chaos engineering alone cannot anticipate.

03

Fragere 2.0: Automated Engineering Pipeline

A framework without implementation is simply a concept. Fragere 2.0 evolves this framework into an automated engineering pipeline, providing a digital platform that automates the STAMP/STPA plus Chaos Engineering framework.



Find Google STPA publications—including books, articles, trainings, and more—for free at sre.google/stpa

STPA (System Theoretic Process Analysis) at Google

Google's SRE team pioneered methods to keep failures rare by engineering reliability into every part of the stack—Service Level Objectives (SLOs), error budgets, isolation strategies, thorough postmortems, progressive rollouts, and other techniques. In the face of increasing system complexity and emerging challenges: what's next? How can we continue to push the boundaries of reliability and safety?

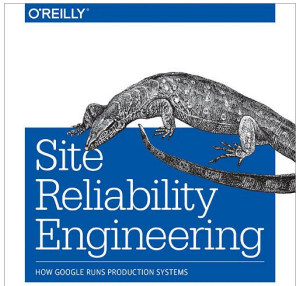
To address these challenges, Google is using System Theoretic Process Analysis (STPA) to analyze pure software systems and discover the unknown unknowns: risks of which you are unaware and not actively seeking. Learn more about how we're using STPA in the following resources.



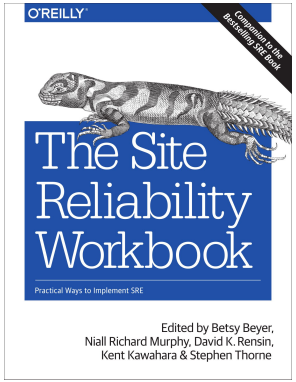
Mapping a Better Future with STPA talk from SREcon Americas 2025



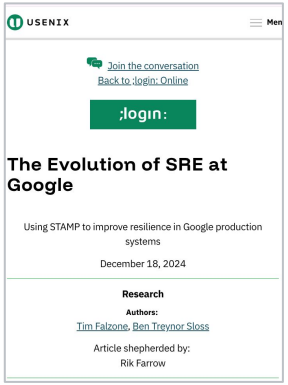
STPA for Software Systems -- Illuminate the Unknown Unknowns SREcon EMEA 2025



Edited by Betsy Beyer, Chris Jones, Jennifer Petoff & Niall Richard Murphy



Edited by Betsy Beyer, Niall Richard Murphy, David K. Rensin, Kent Kawahara & Stephen Thorne



Thank you