

SDXL から AI エージェントへ - クリエイティブ AI プラットフォー ム 3 年間の設計判断 -

Google
Cloud
Next

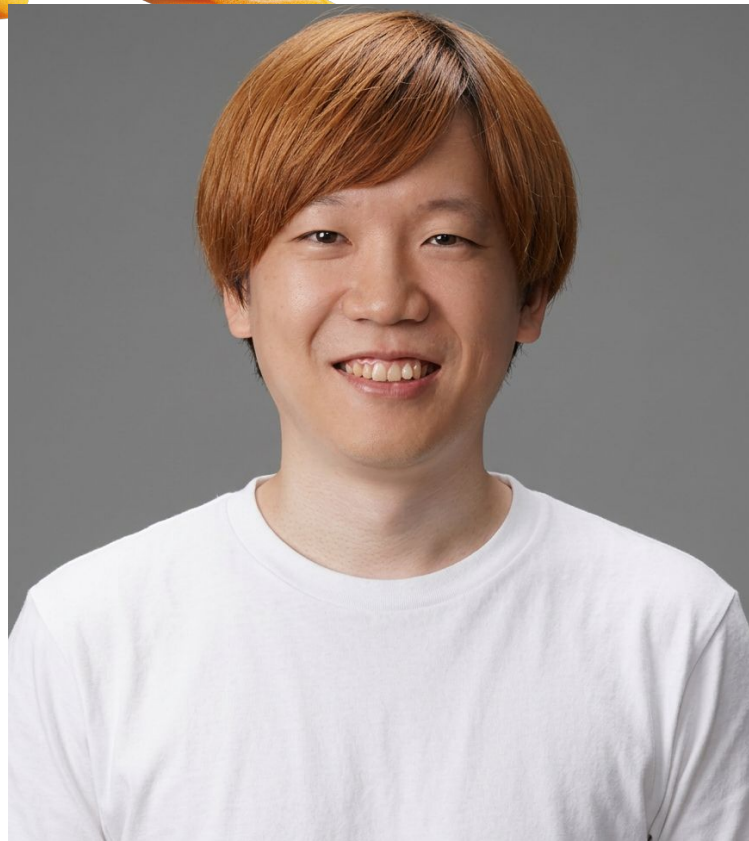
Tokyo





瓜生 和也

株式会社サイバーエージェント
AIクリエイティブカンパニー
AI SCREAM 開発責任者



Agenda



01

AI SCREAM とは何か

3 年で社内に根付いた生成 AI プラットフォーム

02

変えなかった設計

モデル交代を吸収するアーキテクチャ

03

新たに作った設計

3 ヶ月で立ち上げたエージェント基盤

04

運用から見た設計判断

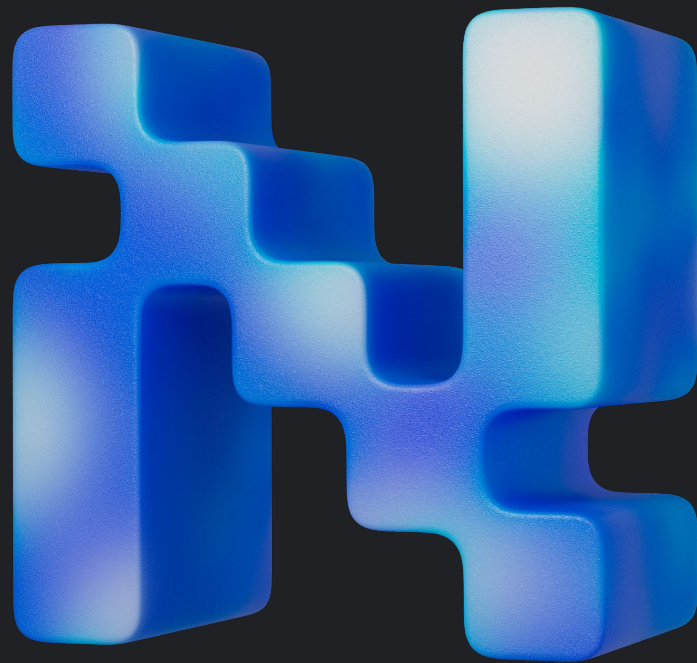
当たった判断 / 選び損ねた判断

05

AI SCREAM のこれから

社内で 3 年支えた仕組みを社外にも

01. AI SCREAM とは何か



生成 AI を社内全体へ

業務利用の 壁

GPU + Python

クリエイター・営業に届かない



社内全体に届けるプラットフォーム

新規事業として 2023/10 に立ち上げ

3年で社内に根付いた

30+

導入モデル数

3年累計

120万

累計生成枚数

2025/02 ~ 2026/02

30%

制作時間短縮

従来比

今日お話すること

Generate 機能

「変えなかった設計」

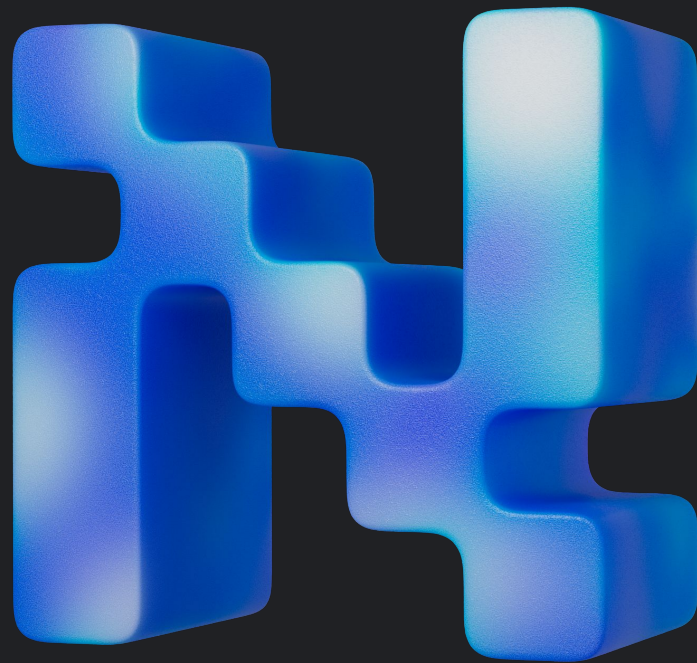
Stable Diffusion XL から API 連携
まで 30 以上のモデル導入を吸収し
てきたアーキテクチャ

Agent 機能

「新たに作った設計」

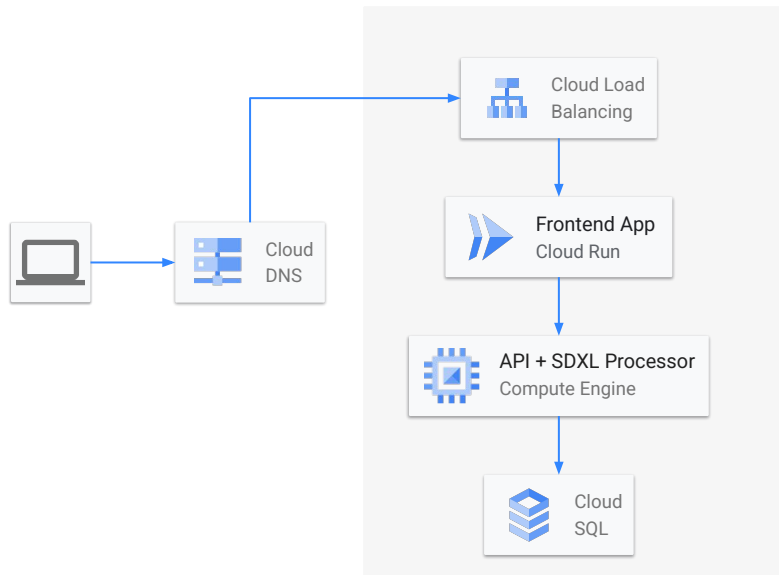
エージェント基盤を 3 ヶ月でリリースし
たアーキテクチャ

02. 変えなかった設計

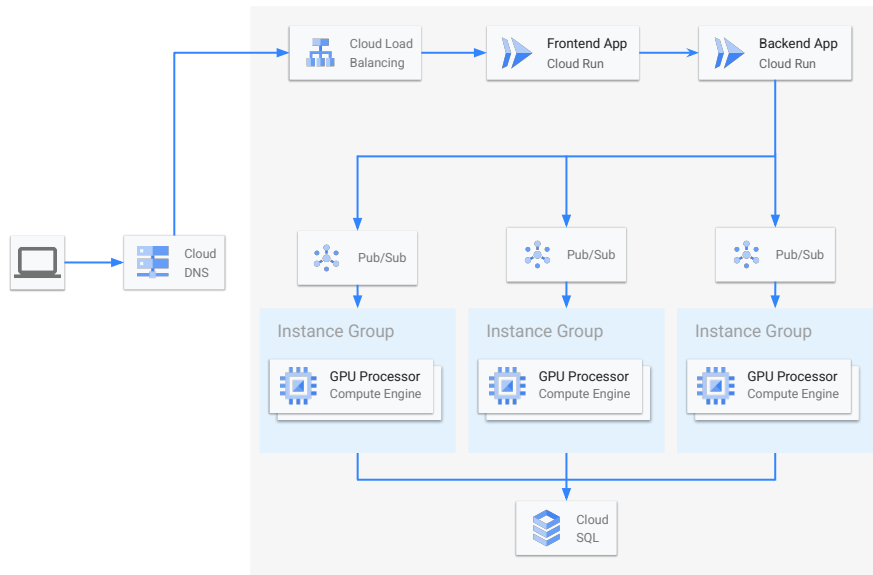


全社展開に向けたリアーキテクチャ

2024/01 ~ 社員 8,000 人中 3 割 (2,400 人) が日常利用するアーキテクチャへ

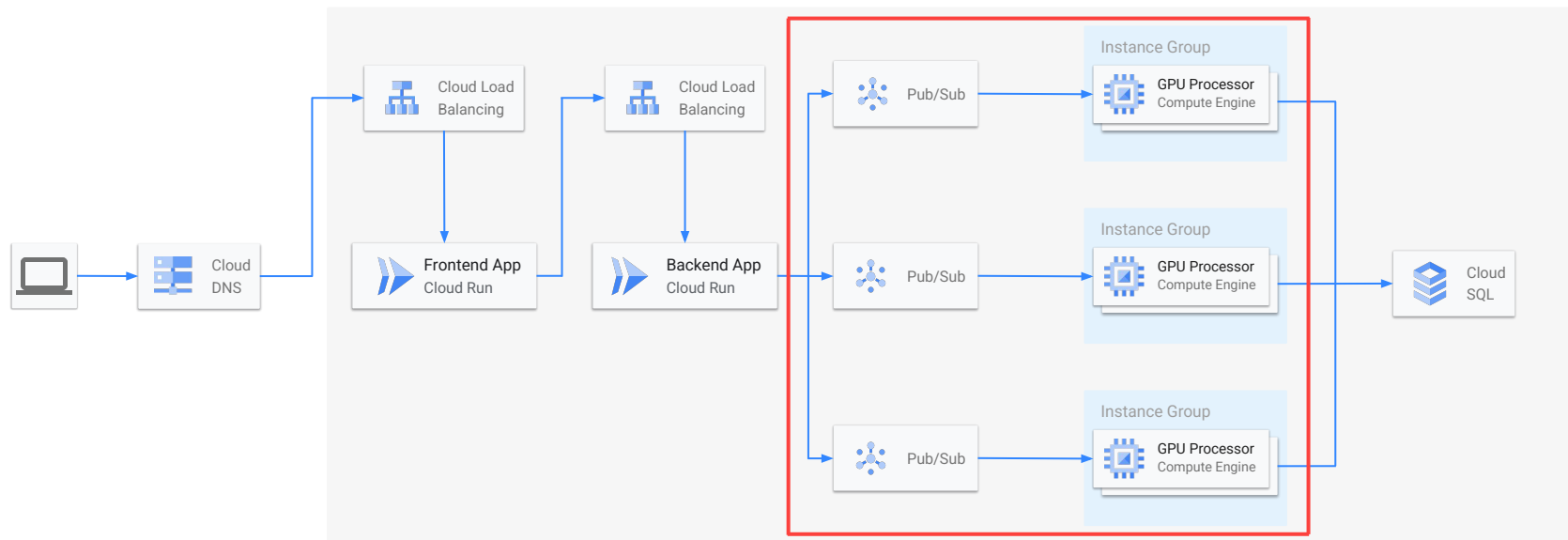


2023/11 ~ Stable Diffusion XL 単体



Pub/Sub による非同期境界・推論ワーカー分離

インフラ層は設計を作り込む



事業要求

社員 8,000 人中 3 割 (2,400 人) が日常利用を想定

物理制約

GPU モデルが大きく 1 VM に乗らない / モデルごとに VM 分離

アプリケーション層は設計を緩める

2 週間に 1 モデル

2024/06 からの目標 - 最速でモデルを提供して常に最新モデルを使えることを価値とする



守る部分

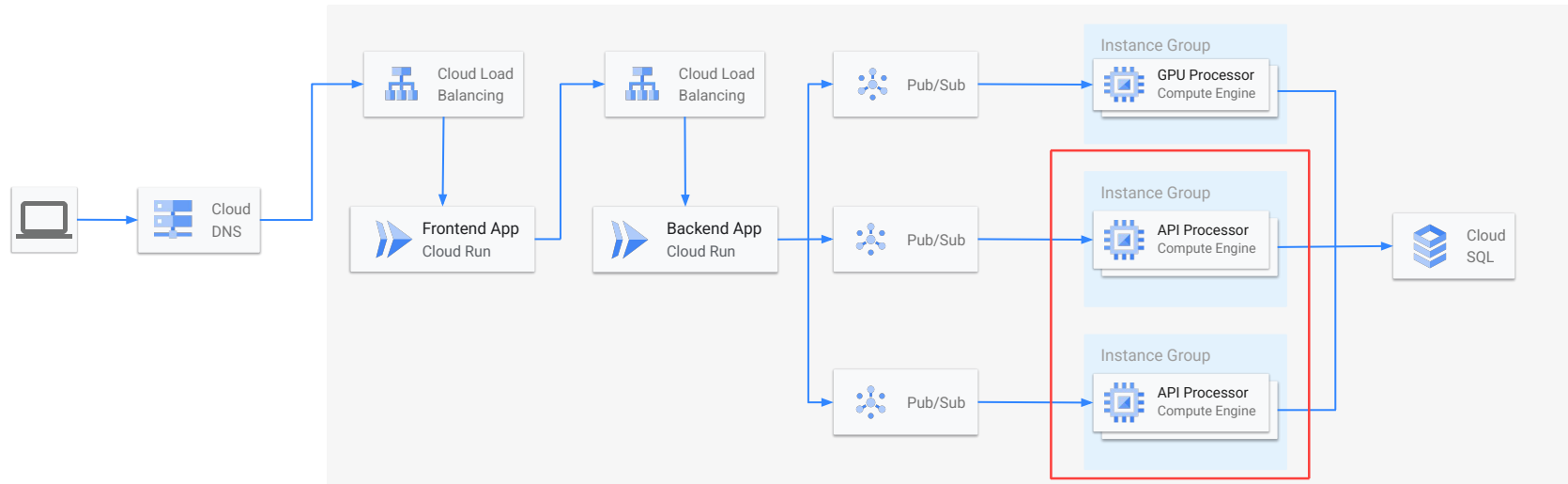
データ設計 / アーキテクチャ設計

緩める部分

推論ワーカー / API 実装 (型化 / コピーを意図的に許容)

API 連携でもアーキテクチャを変えずに吸収

2024/07 ~ Stable Image Core / Ultra で初の API 連携での生成 AI モデル導入



その後、同じ枠組みで吸収

Imagen / Veo / NanoBanana Pro など 30+ モデル

反省 - Cloud Run 移行を逃した

認識していた 2 つのタイミング

Cloud Run worker pools

当時 Push 型のみ

Pub/Sub Pull 型の nack リトライを自前で実装する必要があった

Cloud Run GPU 対応

GPU 推論ワーカーも載せ替え可能に

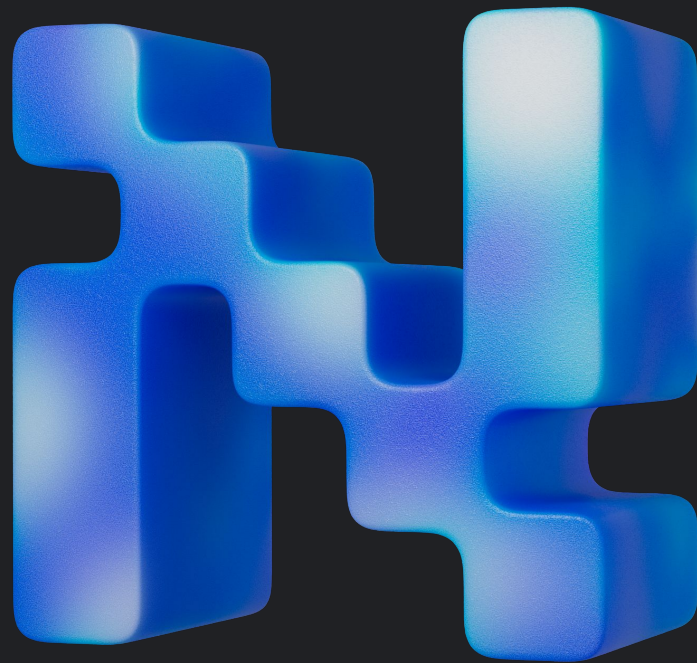
自前 VM 管理を減らせるはずだった

何故踏み切れなかったか

リトライの自前実装の負担 + 並行開発のプレッシャー

→ 結果、自前の VM 管理を続けたまま現在に至る

03. 新たに作った設計



エージェント開発の立ち上がり



Agent Development Kit を選んだ 3 つの理由

Google 製のオープンソース

長期サポートが見込める

A2A プロトコル

Google が推進する流れ

Google Cloud との親和性

マネージド サービスも視野

新規構築と既存資産の再利用

新規構築

エージェント基盤

ADK
Cloud Run
PostgreSQL

別アプリケーションとして最初から設計



呼び出し

既存資産の再利用

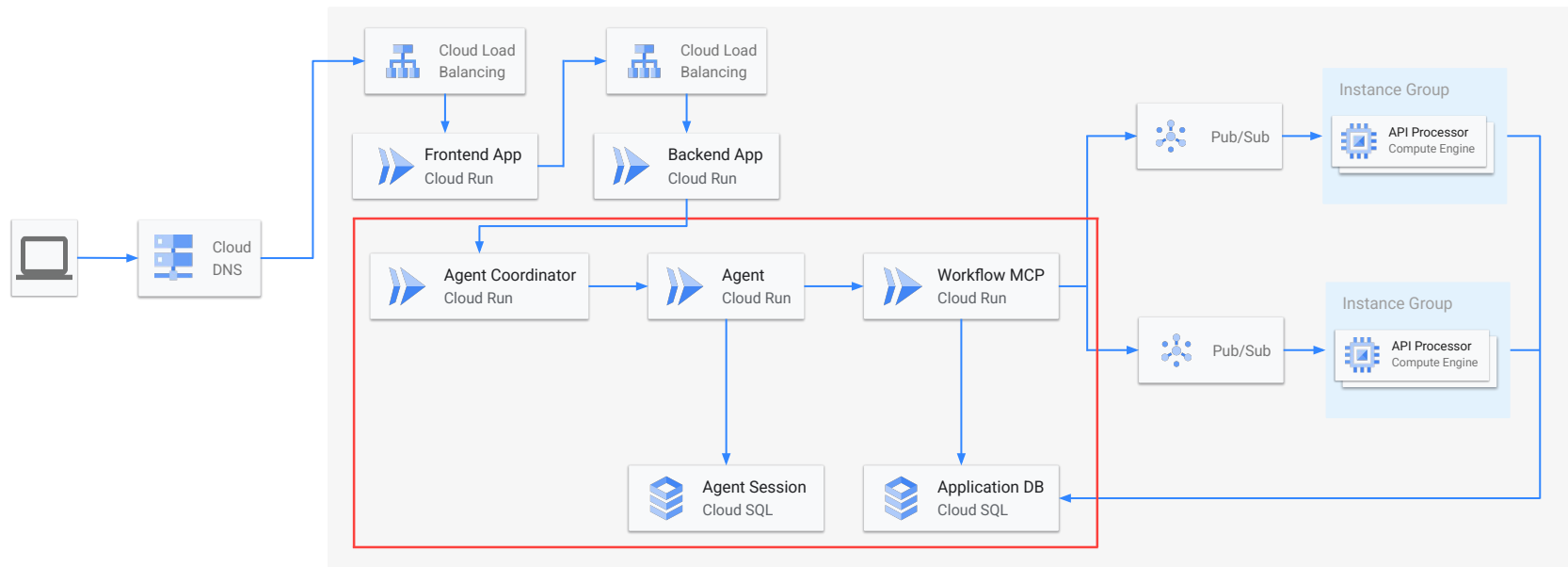
生成基盤

Pub/Sub
推論ワーカー

2章の生成 AI 基盤を呼び出す

全てゼロから開発したわけではない - 再利用すべき抽象を見極めた

アーキテクチャ全体像



Agent Coordinator

SSE 通信を Backend から
切り離す Proxy

Agent + Workflow MCP

ADK エージェントと生成フローの分離

DB 2 系統

Session は専用 PostgreSQL
アプリ DB は既存 PostgreSQL 共有

稼働中のエージェント



画像生成

プロンプトから
画像を生成



ブラッシュ アップ

画像を
範囲指定で差し替え



バナー生成

LP や参考画像をもとに
広告バナーを生成



リサイズ

アスペクト比変更
レイアウト組み換え

現場での使われ方

AI SCREAM Agent3 が画像を生成します

✓ 商品背景を消去 >

✓ パターン1 >

✓ パターン2 >

✓ パターン3 >

✓ パターン4 >

✓ パターン5 >

✓ パターン6 >

✓ 生成結果 >



約 20 %

ダウンロード率

生成枚数: 約 10,000 枚
ダウンロード数: 約 2,000 枚

生成結果

この生成結果を評価



プレビュー期プロダクトとの向き合い方

①

知って取ったリスク

Gemini Enterprise Agent Platform Agent Runtime (以降 Agent Runtime) + Gemini Enterprise Agent Platform Session Service がプレビューだと把握したうえで採用

②

ドキュメント確認漏れ

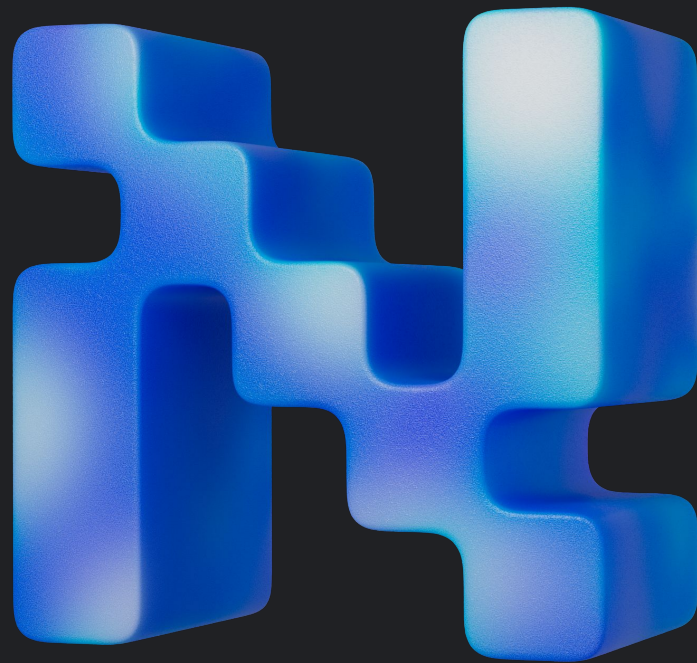
Agent Development Kit もプレビューだった
OSS リリースから GA だと思い込み、開発中にドキュメントで気づいた

③

フィジビリティ確認漏れ

Agent Runtime → Cloud SQL の VPC 非対応
事前検証で確認できていなかった制約として、デプロイ時の動作確認で発覚
→ 2025/07/17 に Cloud Run へ即決移行 (1 人 / 2 日)、2025/07/22 リリース

04. 運用から見た設計判断



同じ判断軸で都度選んだ結果

当たった

選び損ねた

2 章

Pub/Sub 抽象化

全社展開 / GPU 物理制約に対応

Cloud Run 移行を逃した

認識していたが踏み切れず

3 章

エージェント新規開発 + 生成再利用

再利用すべき抽象を見極めた

プレビュー期の事前確認漏れ

ADK プレビュー / VPC 非対応

事業要求 x 物理制約で都度選んだ

3 年の運用から見えた 3 点

①

抽象化のレベルは「事業要求 x 物理制約」で都度判断する

完璧な抽象化は目指さず、その時点で必要な粒度を判断する

②

抽象化したら「短期の負担」と「長期の効果」を分けて意識する

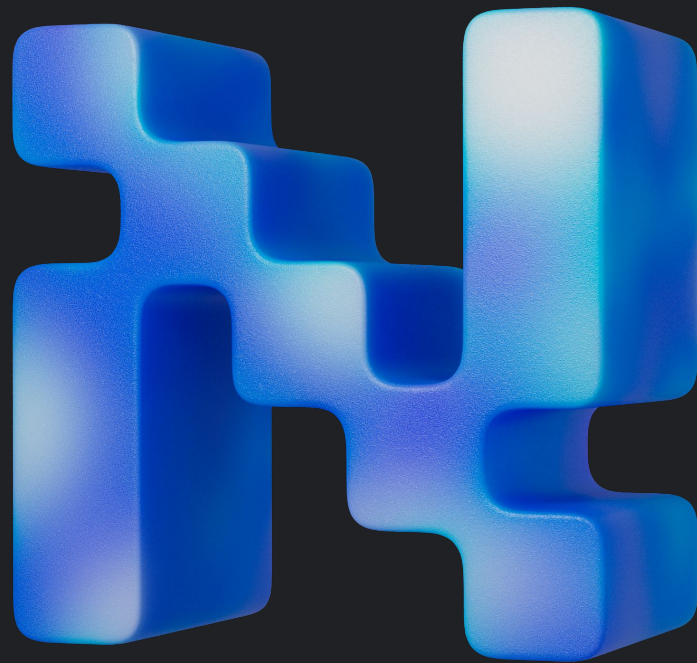
認識していても踏み切れないと意味がない。戦略が戦術を喰う罠を避ける

③

プレビュー期プロダクトは 3 層で整理する

受容済みリスク / ドキュメント確認 / フィジビリティ確認

05. AI SCREAM のこれから



社内で 3 年支えた仕組みを社外にも

3 年支えてきた

Generate 機能

静止画生成

動画生成

Agent 機能

バナー生成

ブラッシュ アップ



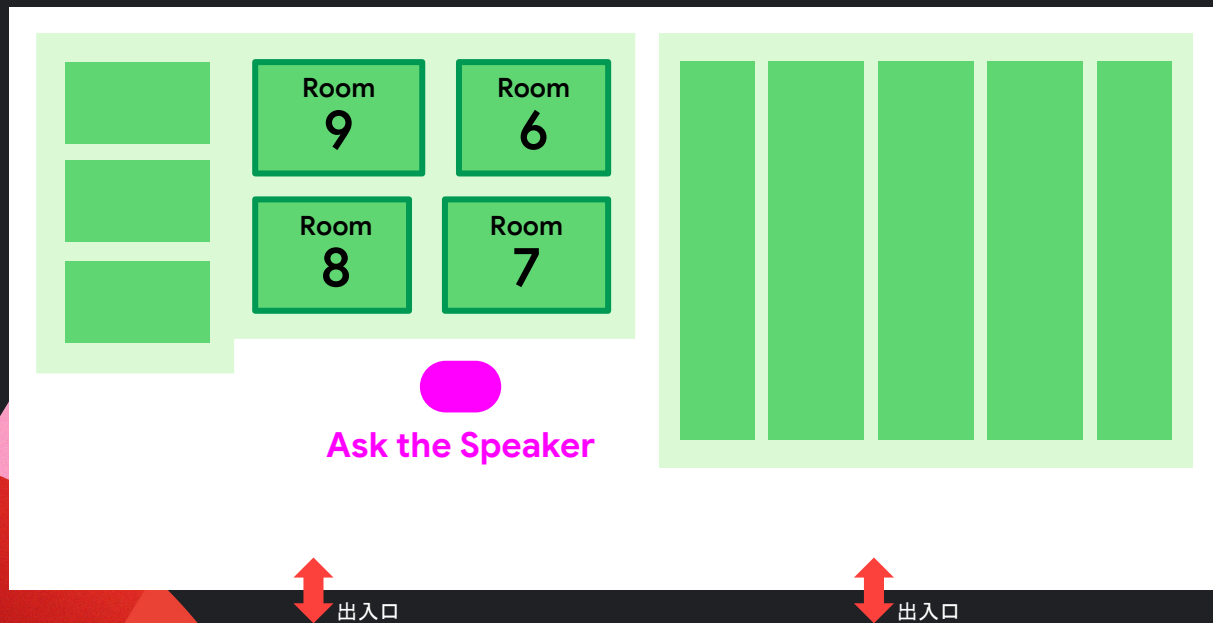
社外にも届けていく

クリエイティブ制作に課題を感じている方、 Ask the Speaker でお話しましょう



Ask the Speaker にぜひお越しください（会場は 4F）

セッションに関する質問にスピーカーが直接お答えします！



Thank you

Thank you

Google
Cloud
Next

Tokyo

