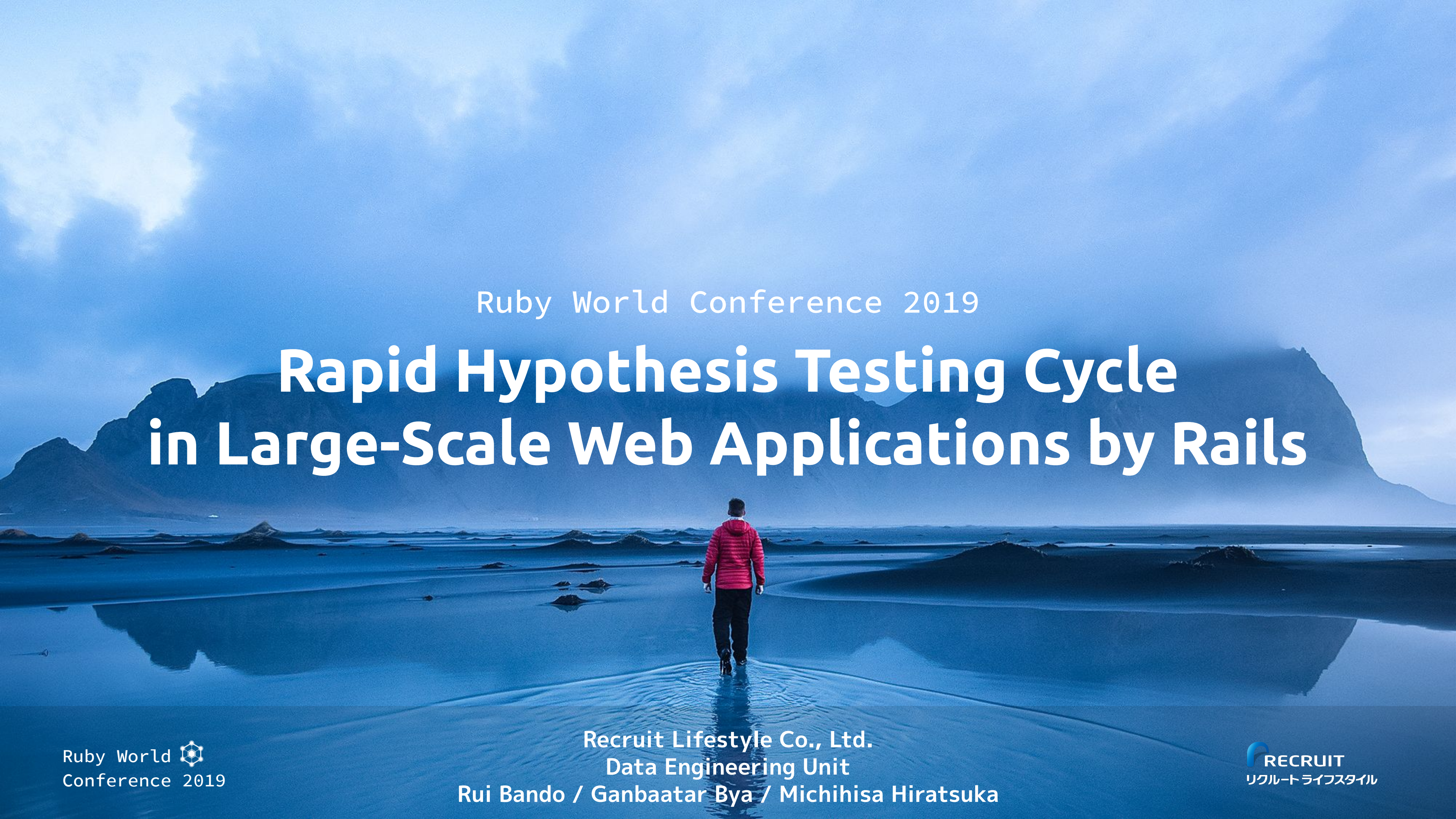


Ruby World Conference 2019

Rapid Hypothesis Testing Cycle in Large-Scale Web Applications by Rails

Recruit Lifestyle Co., Ltd.
Data Engineering Unit
Rui Bando / Ganbaatar Bya / Michihisa Hiratsuka

Ruby World Conference 2019

RECRUIT
リクルートライフスタイル

Speaker

Rui Bando

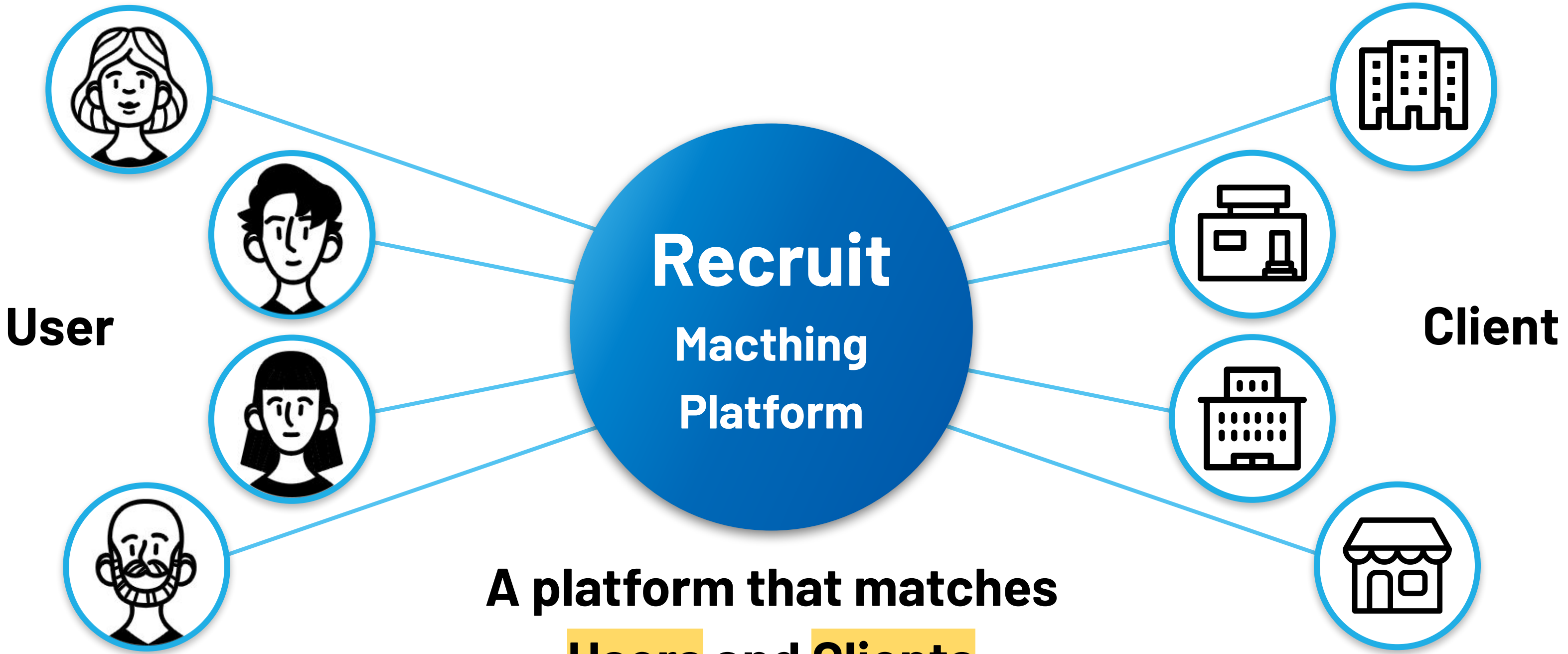
Recruit Lifestyle Co., Ltd.
Data Engineering Unit
Product Manager, Data Planner

Ganbaatar Bya

Recruit Lifestyle Co., Ltd.
Data Engineering Unit
Engineer

Our Company

Our Business



A platform that matches
Users and Clients

Our services

じゃらん じゃらんゴルフ エイビーロード AB-ROAD HOT PEPPER HOT PEPPER グルメ HOT PEPPER Beauty TABROOM ポンパレモール ブッキングテーブル 人間ドックのここカラダ 保険チャンネル seem

RECRUIT
リクルートライフスタイル

AirREGI AirPAY AirRESERVE AirWAIT AirMATE AirREGI ハンディ AirSHIFT トリップAIコンシェルジュ Restaurant BOARD SALON BOARD

About our company

Biz segments	Region	Main Biz	Group companies
HR Technology	HR Technology		
Media & Solutions	Marketing Solutions	Housing and Real Estate	RSC
		Bridal	RMP
		Travel	RLS RECRUIT リクルートライフスタイル
		Dining	
		Beauty	
	HR Solutions	Recruting in Japan	RCA・RJB
Staffing	Japan Operations		
	Overseas Operations		

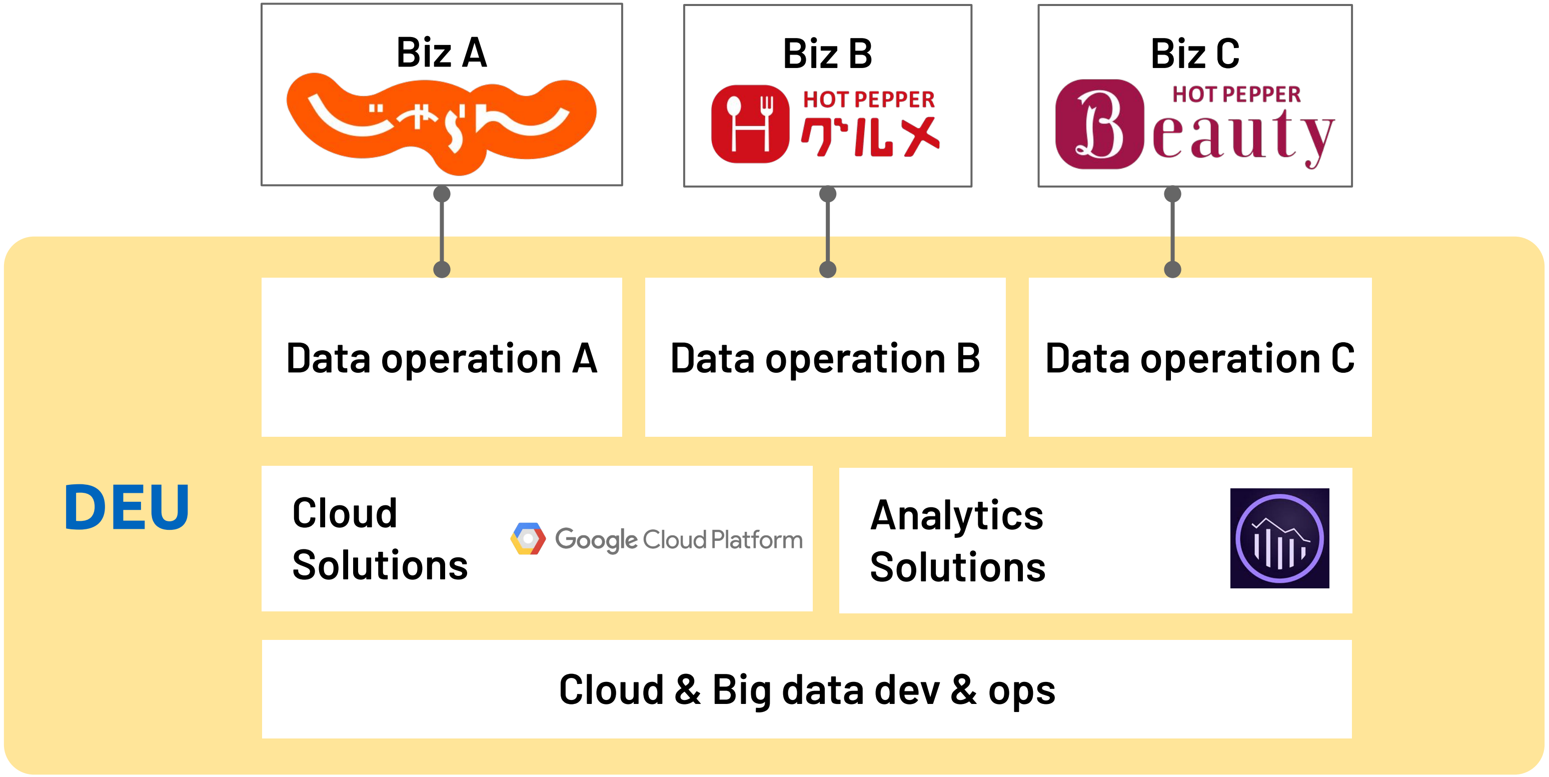
Trading Volume March 2019

Approx.180 Billion JPY

Data Engineering Unit

Crossing dept. that develop and operate data

Data Engineering Unit’s mission



Our Data Solutions

Item recommendation

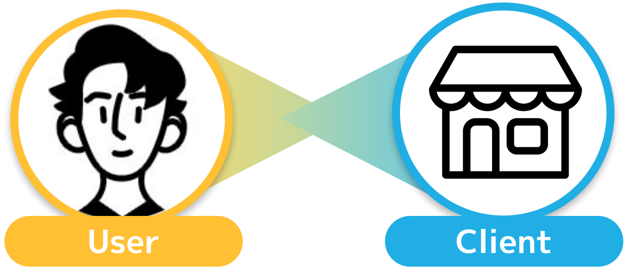
Point distribution optimisation

Preference estimation

Sort items

Real time offer

Revenue Simulation



RFM / LTV model

TV-CM effect estimate

Separation prevention

Chat bot

Document and image classification

Price optimization

Plan recommendation

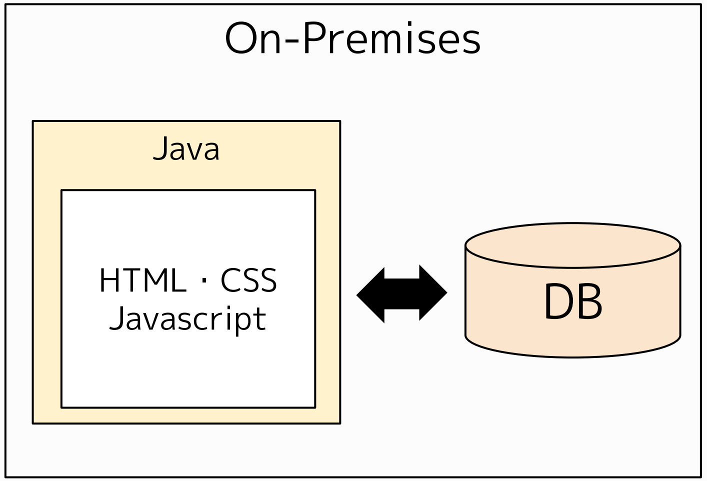
Etc.

Our action

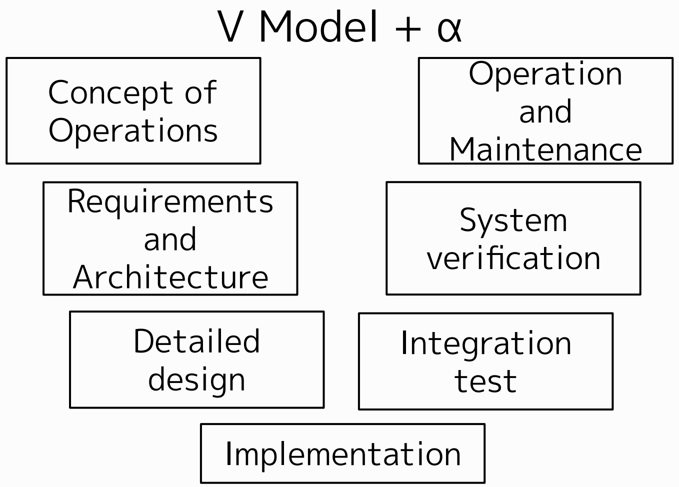
Our History

Current status of Large-scale service over 10 years

Architecture



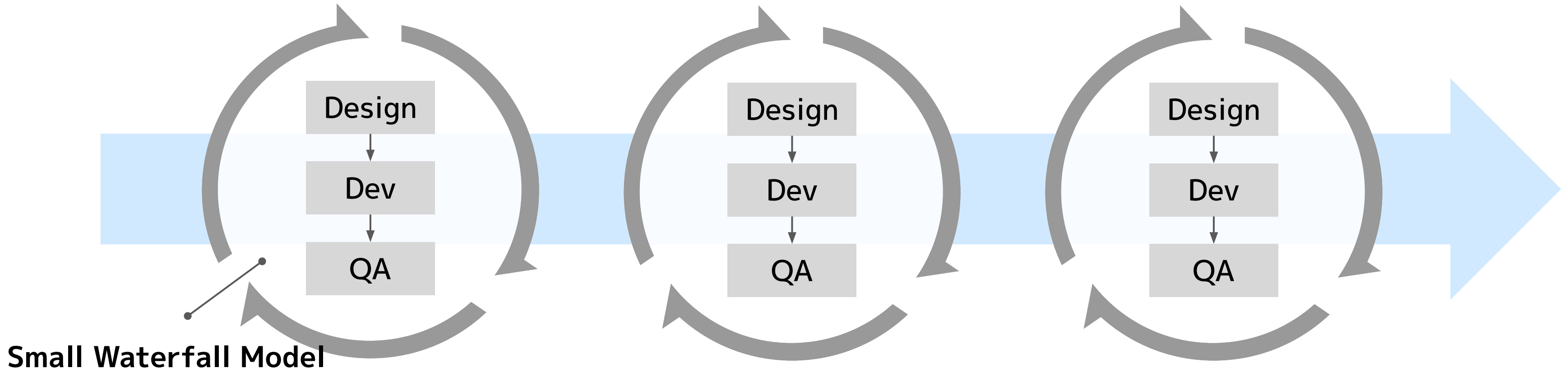
Dev Process



Large-scale services require reliability through stable operation
※Great impact when a problem occurs

Change Development Process

Process improvement including Agile in "Our service"



Architecture is also On-Premises

Existing architectures face difficult challenges

Cloud and our architecture

Focus on new architecture such as Cloud and serverless

- On-Premises to Cloud
- New architecture such as serverless has appeared

Migrating large services is not easy

- Understanding migration is really a challenge.
- Migration cost is comparable with new cloud development services.
- The direction of product is not affected by the keyword "Cloud"

Understand what our users really want!!

For Value Provision

The importance of “Approach method” is increasing

「P：people」「P：process」「T：technology」

Understand these keywords.

↓

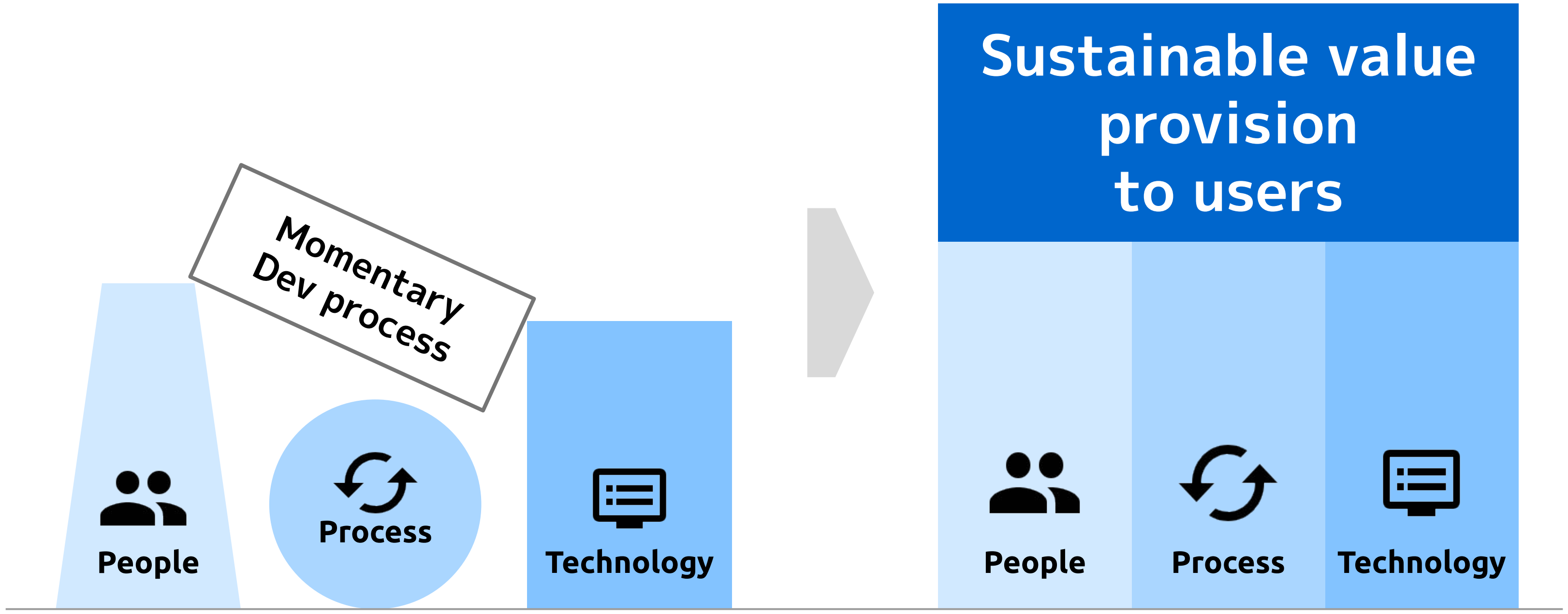
Focus on providing valuable services to users

Understand importance of **「How」**

Product value is more important to user than architecture

※Don’t be the developer’s self-satisfaction

Anti-pattern : Unbalanced P-P-T



A momentary architecture and development process
do not lead to sustainable value provision

Utilizing tech that prioritizes value provision

No growth of product just by “Protection”

Actively adopt new technologies such as cloud

Maximize value provision to users

Our Service ✕ New Technology

Hybrid efforts in architecture, dev process and operation.

Understand what is needed now
without considering new tech as the top priority

Separation of “Agility” and “Reliability”

The important thing is "Give value to user first”

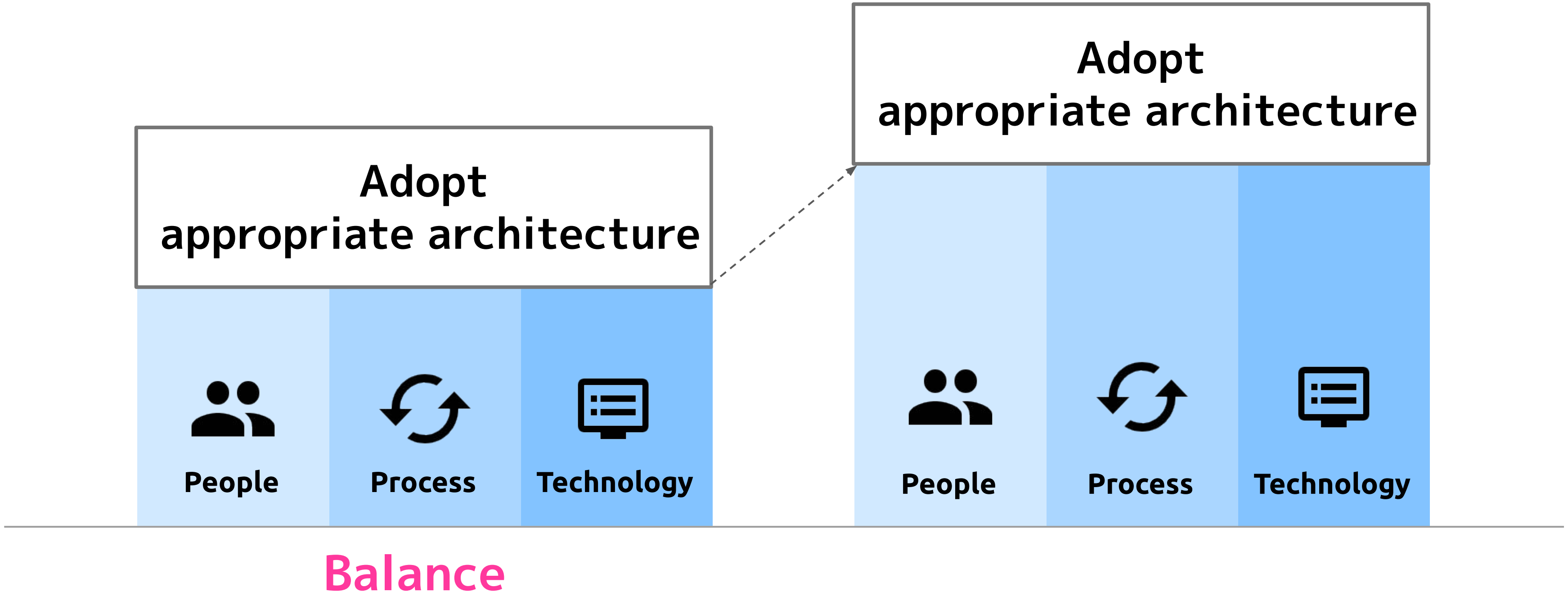
Think about what you should prioritize.
Work on carving out the architecture by considering on an “Agility” and “Reliability” system

Bimodal IT（SoR/SoE）

Reliability	Mode 1 Mode 2	Agility
Waterfall, V-model		Agile, Kanban
Plan-driven,approval-based		continuous,process-based
Long(months)		Short(weeks, days)

Adopting P-P-T balanced architecture

The adoption of optimal tech raise the level of P-P-T


Adopt appropriate architecture
People
Process
Technology
Balance
Adopt appropriate architecture
People
Process
Technology

Our challenge for user

Always thinking about “How” to realize value provision

- Users do not reach without proper “How”
- Depending on your ”How”, the speed to reach users will vary greatly

Awareness of user and business value.

In architecture, development process and operation,
Think of “Reliability” and ”Agility” separately,
Not aiming to adopt all new programing languages and architectures

Always thinking to pursue user value provision

Dev. Side

Today’s Message

Product Mission

Focus on users

From **developer** viewpoint

Reduce development concerns

Our Strategies

MVP development

Q: How can we execute many testings for users?

A: Just quickly develop features & test it.

No. There are too many concerns!

Difficulties of MVP in our company

Rapid Delivery

+

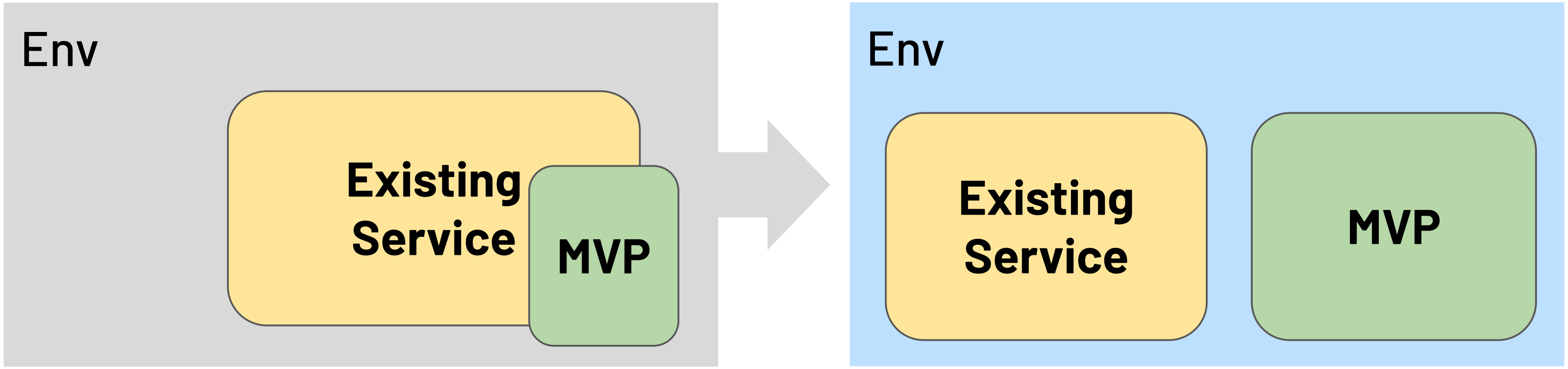
Existing Tech Debt

Expected Quality

High Reliability

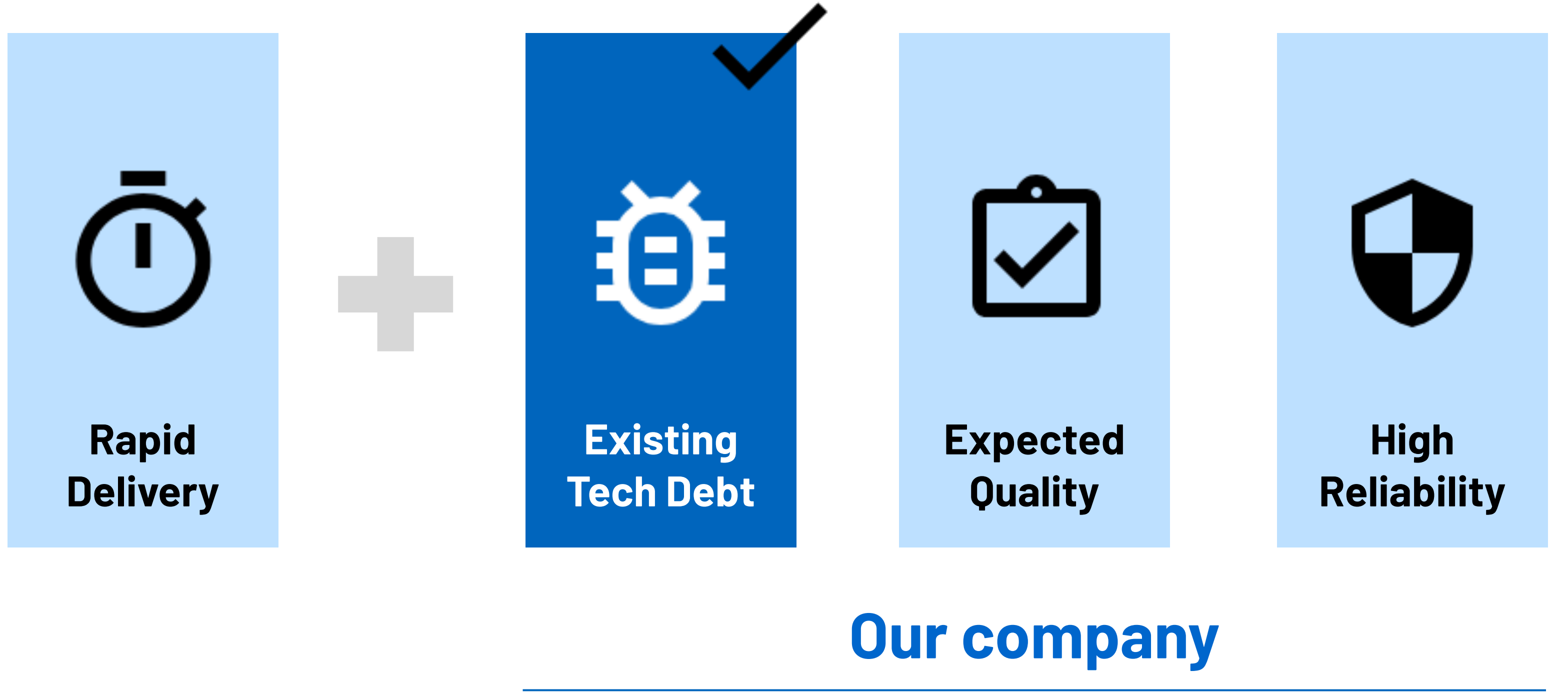
Our company

Strategy 1-1: Separated Environment

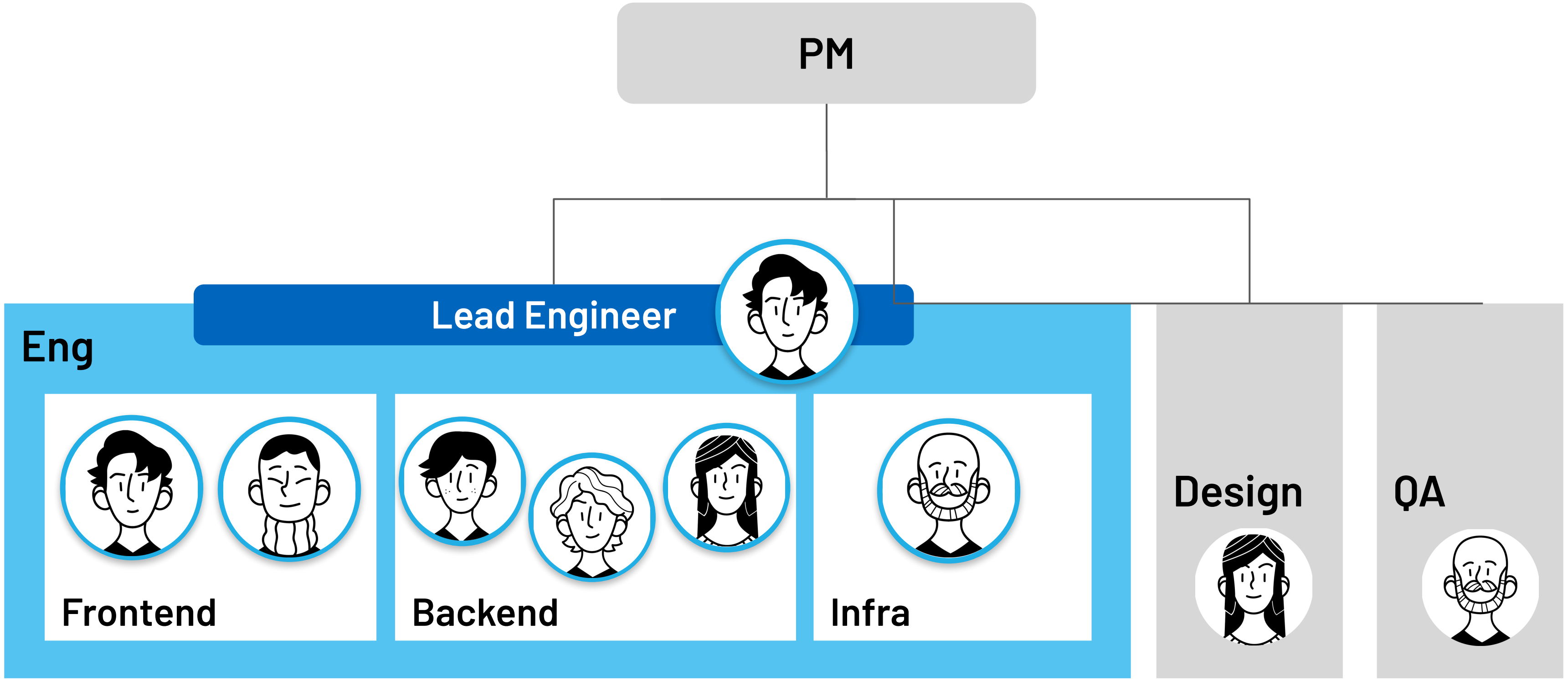


✓ independant on tech. debt
✓ increase # of testings

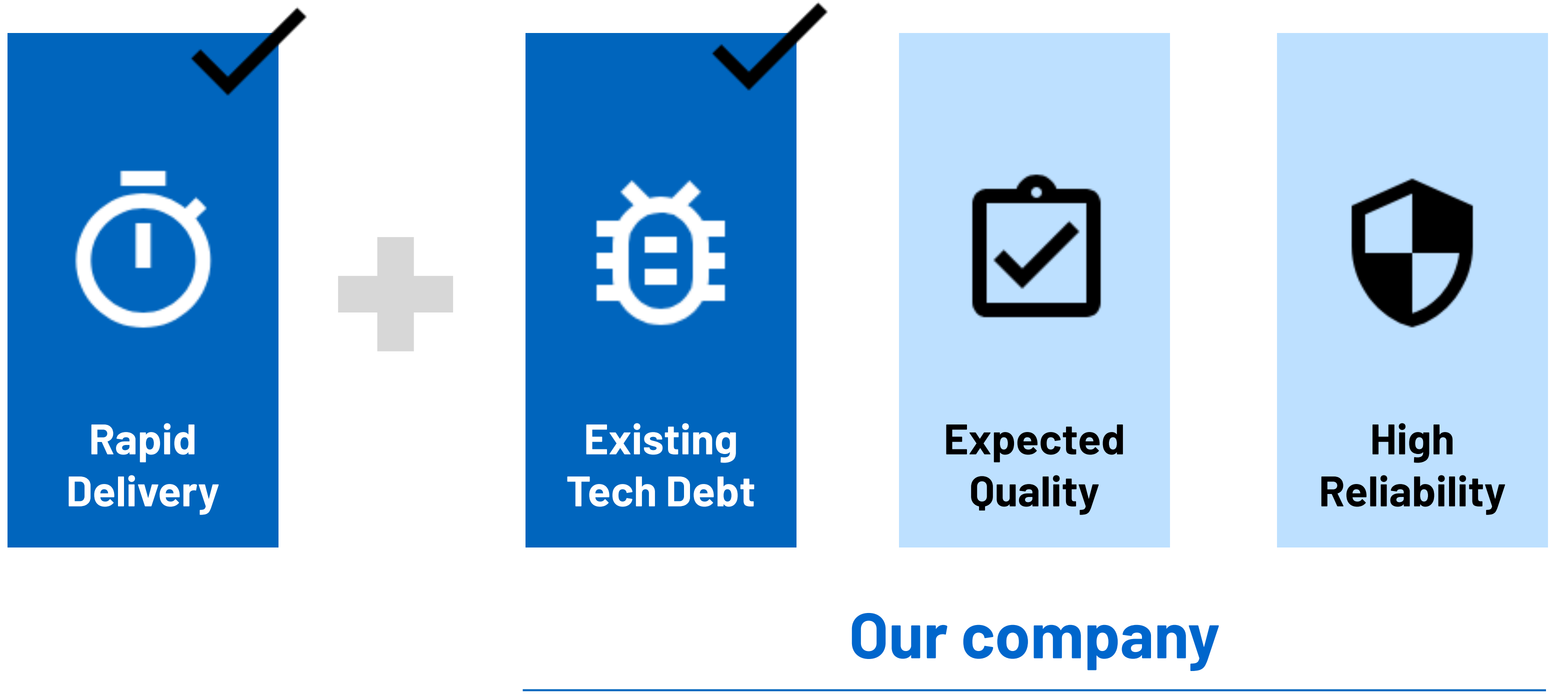
Strategy 1-1: Separated Environment



Strategy 1-2: Independent Dev Team

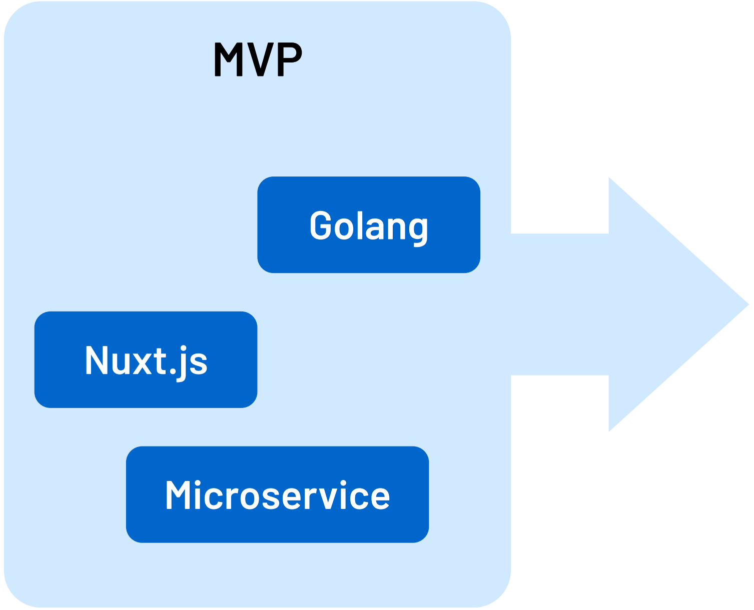


Strategy 1-2: Independent Dev Team



Failures in Architecture

Tech Stack



Difficulties

- 😨 Monitoring
- 😨 Member assign
- 😨 Versioning up
- 😨 Googlability

Failures in Architecture

Tech Stack

MVP

Golang

Nuxt.js

Microservice

Difficulties

Monitoring

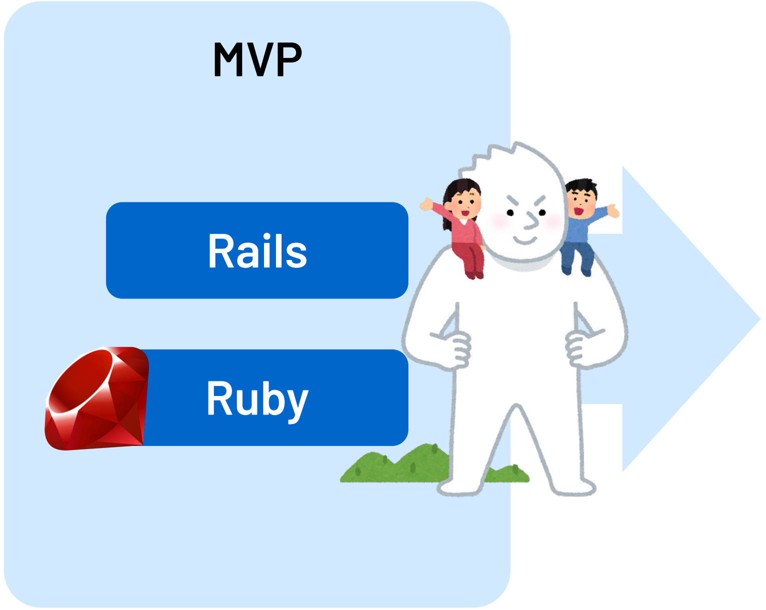
Member assign

Versioning up

Googlability

Is it MVP???

Strategy 2: Ruby on Rails



Benefits

- ✓ Easy for beginners
- ✓ Stable Framework
- ✓ Matured Ecosystems

Strategy 2: Ruby on Rails

✓ Rapid Delivery

+

✓ Existing Tech Debt

✓ Expected Quality

High Reliability

Our company

Strategy 3-1: + Cloud

Tech Stack

AWS & GCP

aws

MVP

Google Cloud Platform

Benefits

✓ More Functionality

✓ More Flexibility

✓ Managed Monitoring

Strategy 3-2: Kubernetes

Tech Stack

AWS & GCP

MVP

Benefits

- ✓ Auto Healing
- ✓ Auto Scaling
- ✓ Rapid Rollback

Strategy 3: Cloud & Kubernetes

✓ Rapid Delivery

+

✓ Existing Tech Debt

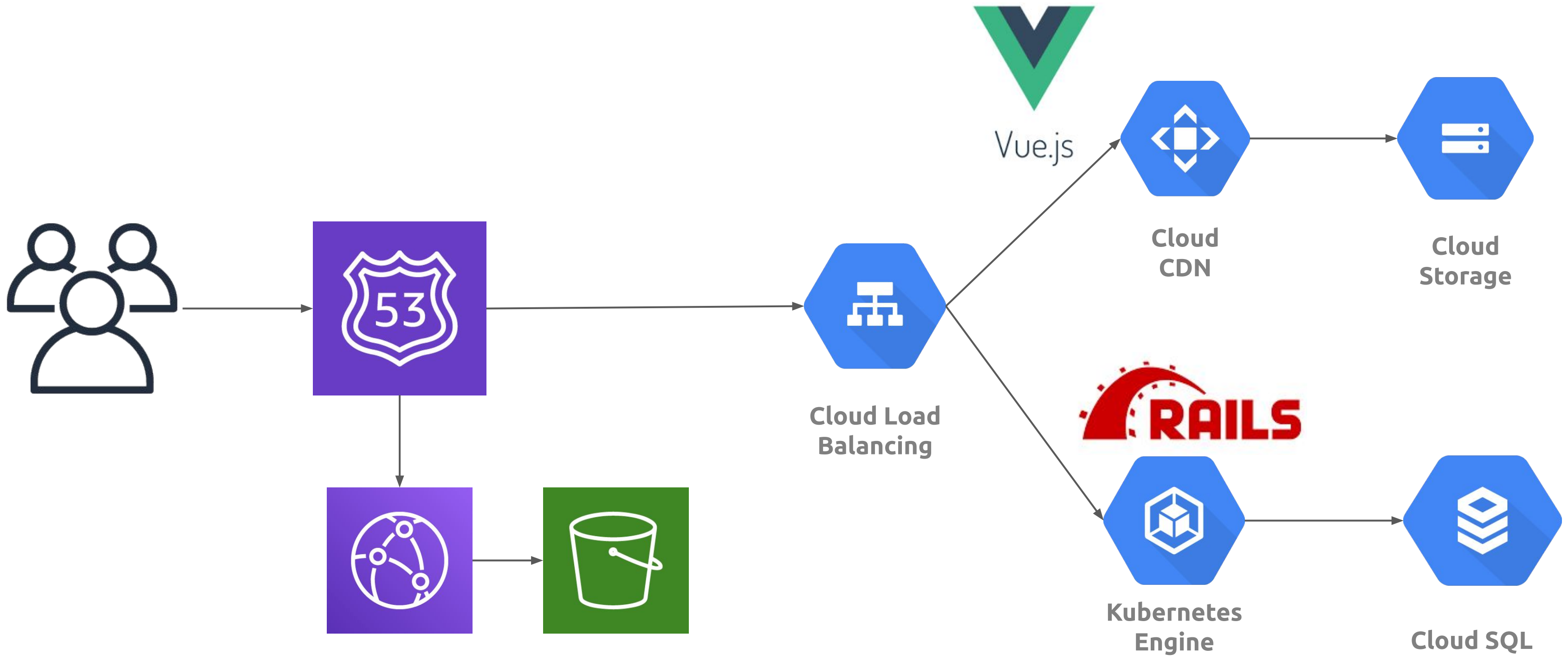
✓ Expected Quality

✓ High Reliability

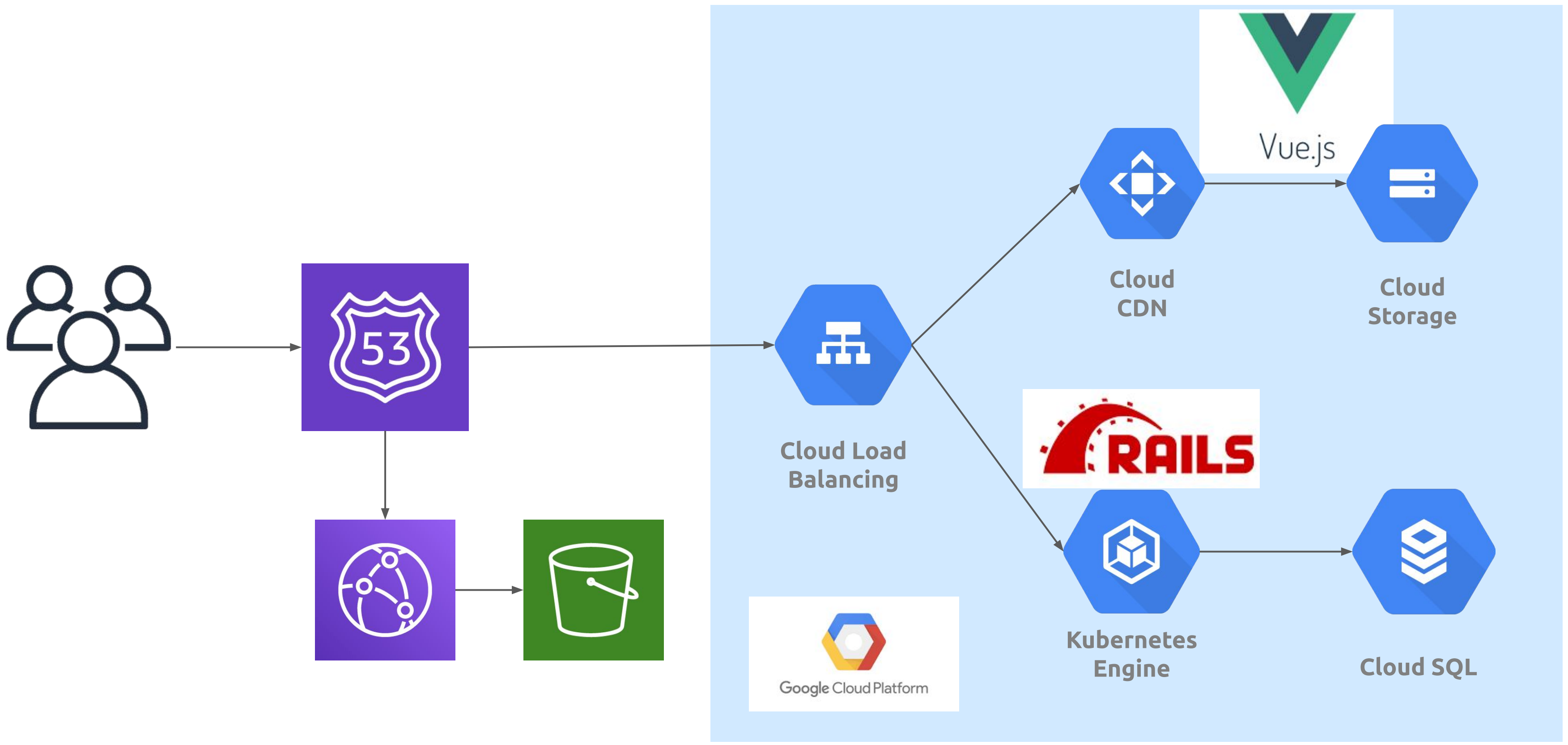
Our company

More Reliability

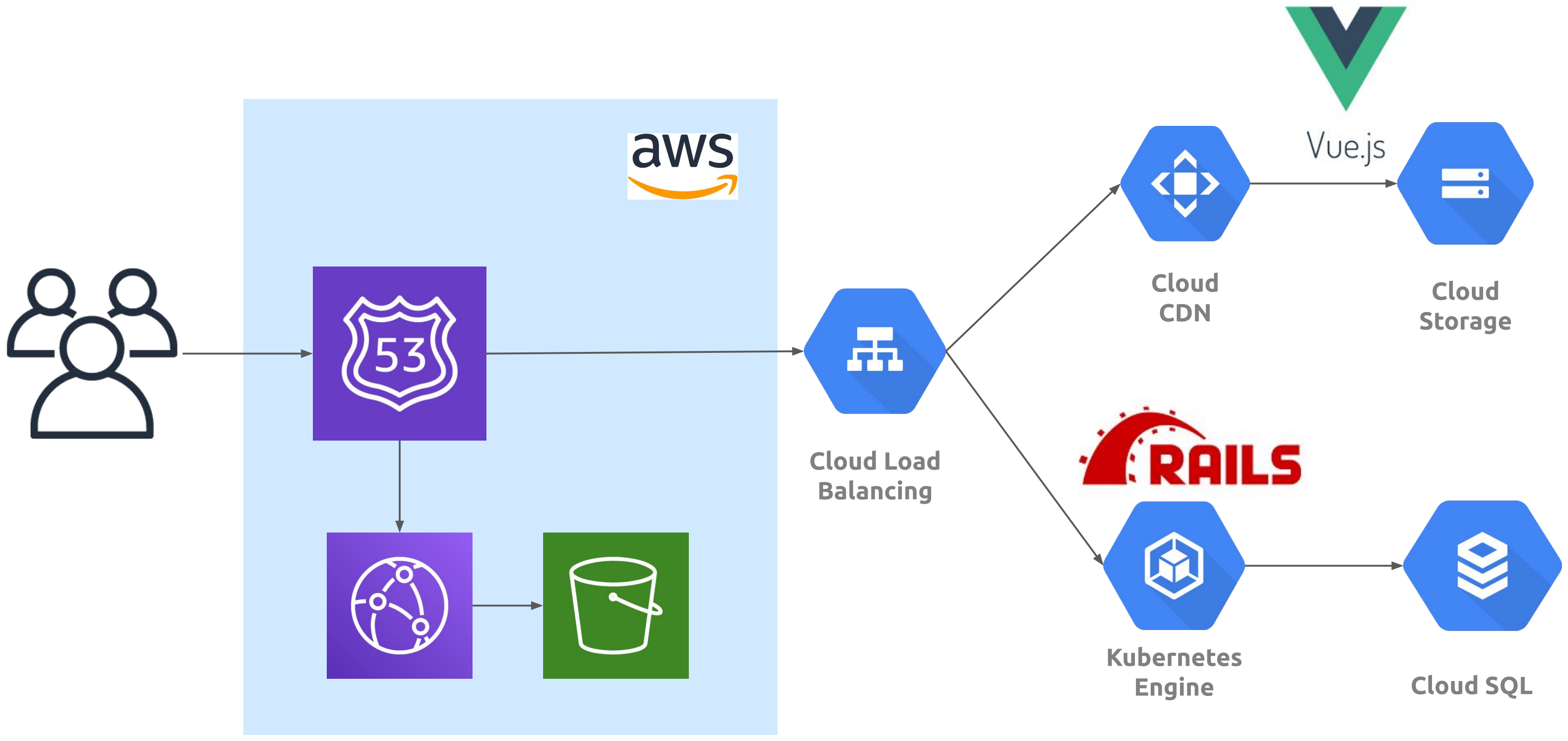
Overview


Vue.js
Cloud CDN
Cloud Storage
53
Cloud Load Balancing
RAILS
Kubernetes Engine
Cloud SQL

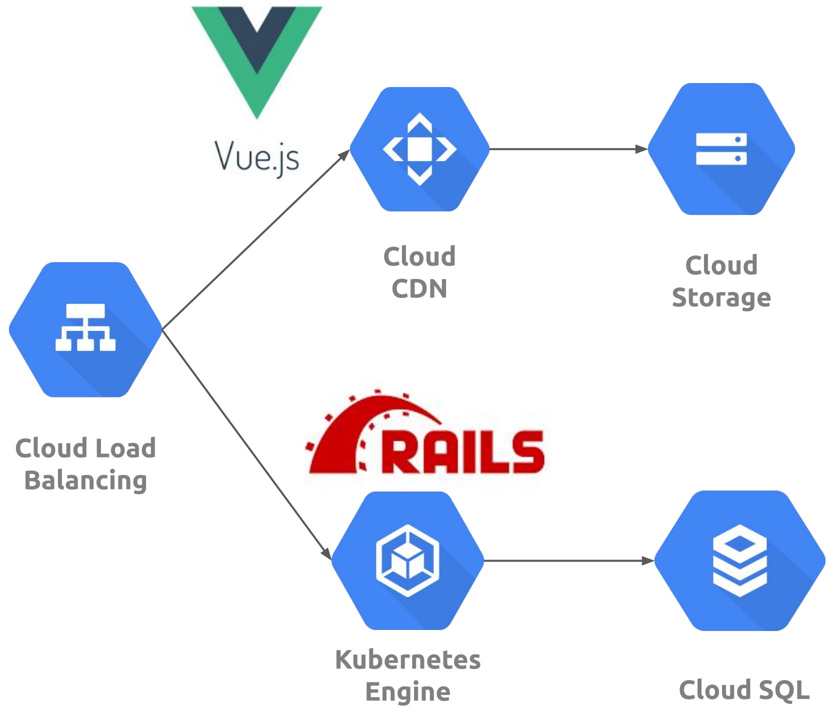
Conventional parts



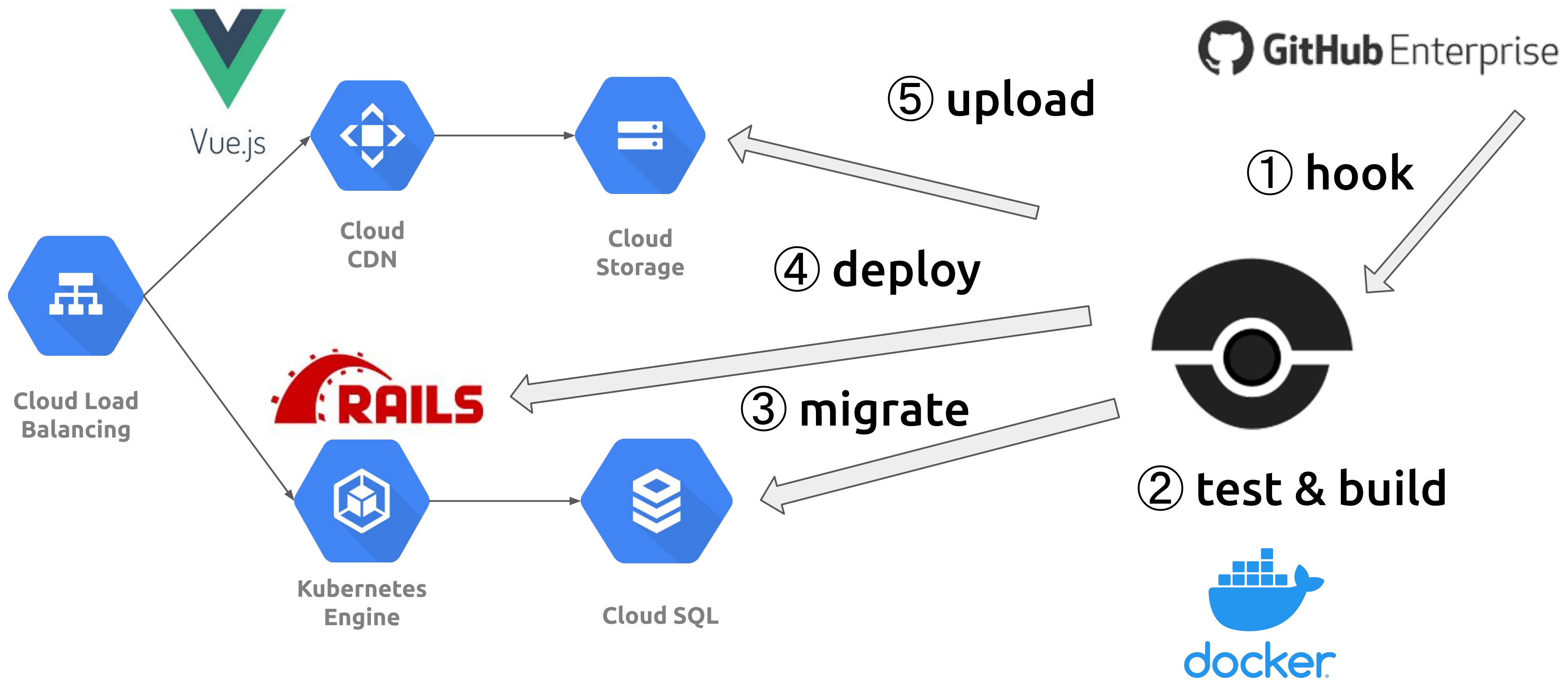
1. Failover


aws
53
Cloud Load Balancing
Cloud CDN
Vue.js
Cloud Storage
RAILS
Kubernetes Engine
Cloud SQL

2. DevOps


Vue.js
Cloud CDN
Cloud Storage
Cloud Load Balancing
RAILS
Kubernetes Engine
Cloud SQL

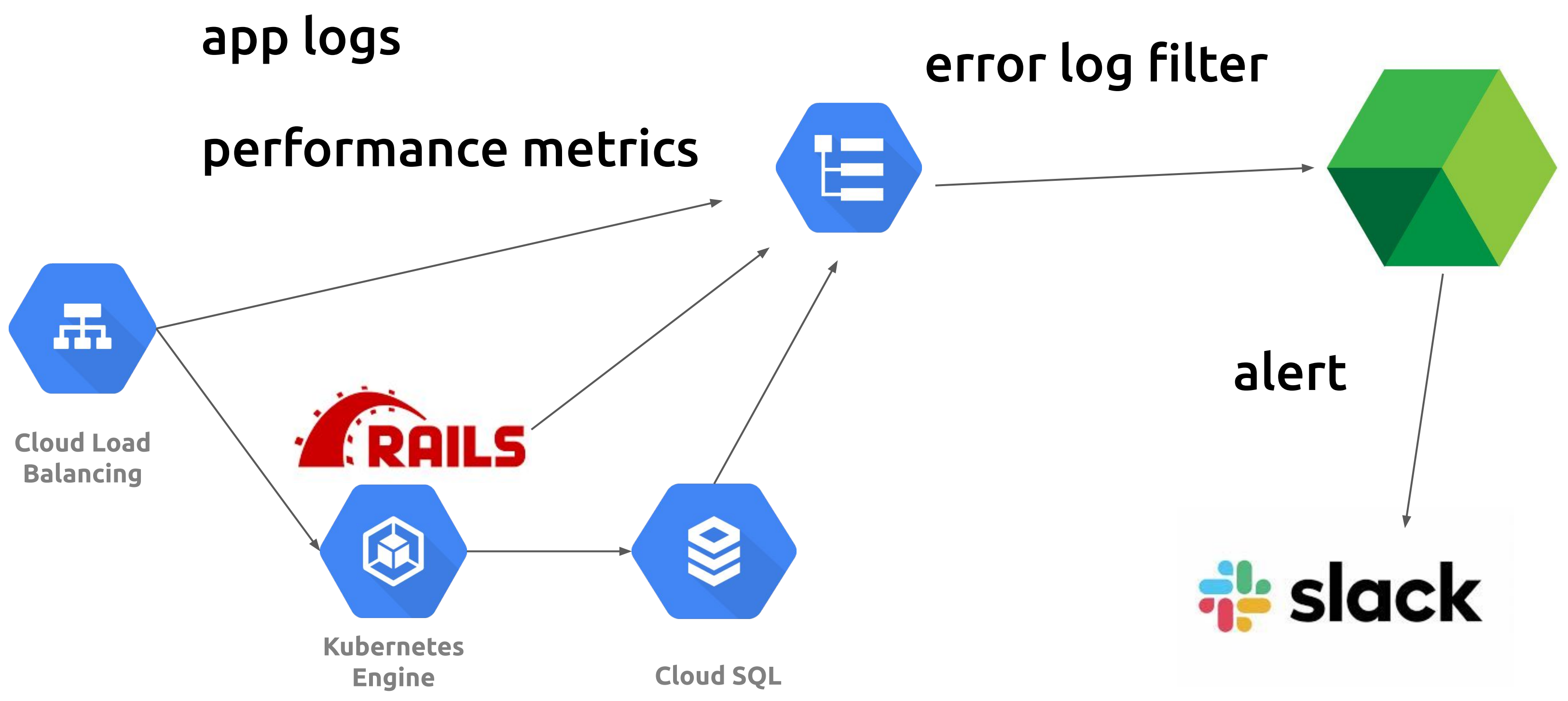
2. DevOps



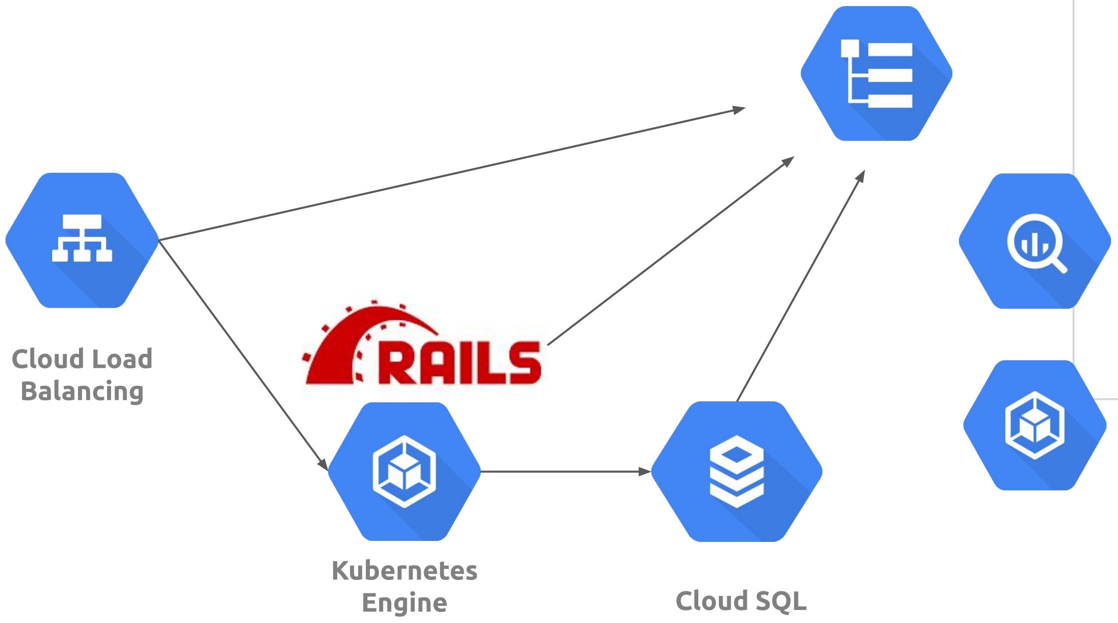
Vue.js
Cloud CDN
Cloud Storage
Cloud Load Balancing
RAILS
Kubernetes Engine
Cloud SQL
GitHub Enterprise
⑤ upload
① hook
④ deploy
③ migrate
② test & build
docker

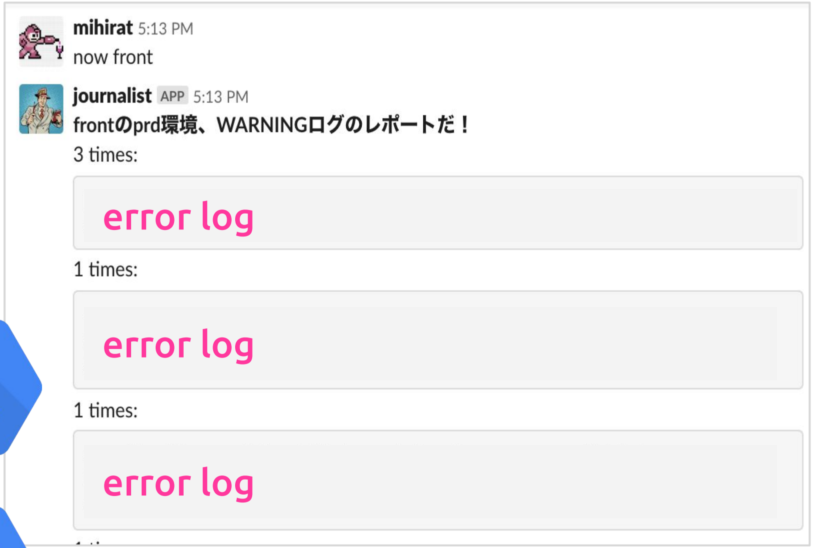
More Quality

Monitoring


app logs
performance metrics
error log filter
alert
Cloud Load Balancing
RAILS
Kubernetes Engine
Cloud SQL
slack

1. Easy Error Log explorer




mihirat 5:13 PM
now front

journalist APP 5:13 PM
frontのprd環境、WARNINGログのレポートだ！
3 times:

error log

1 times:

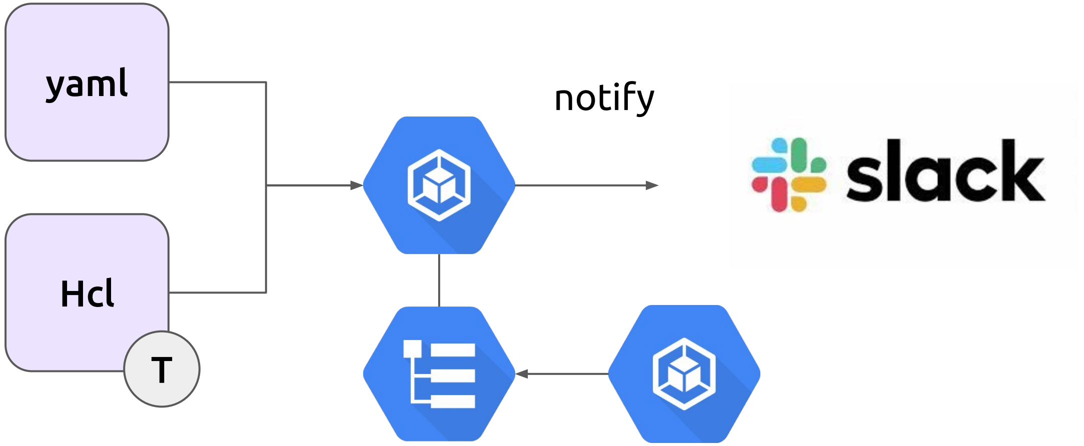
error log

1 times:

error log

2. Codified Alerts

- To manage app alerts, codified settings in Terraform
  - sharable knowledge
  - less human error: not to let errors be ignored



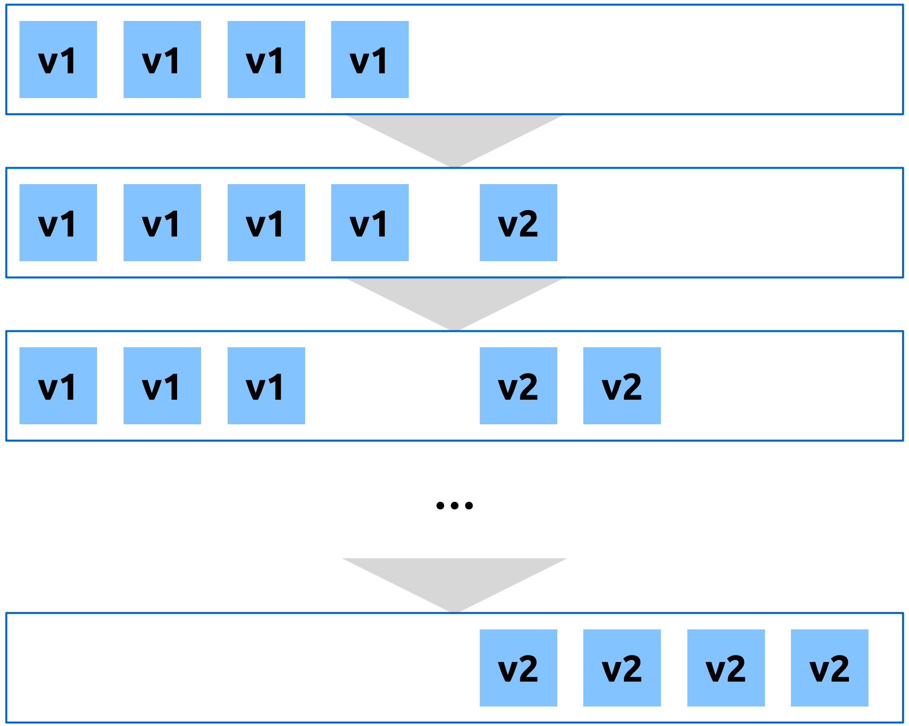
Challenges

k8s pod lifecycle: Rolling Update

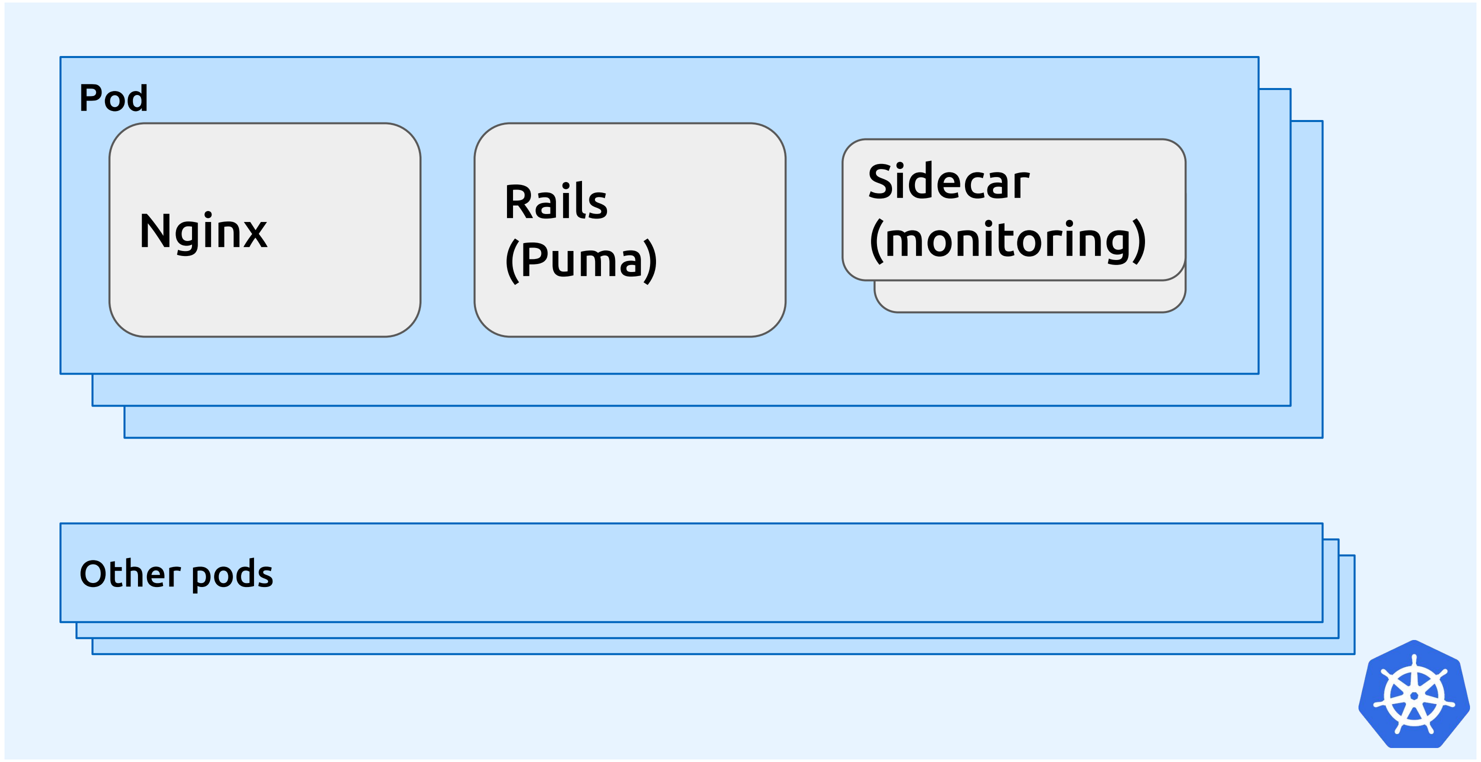
1. Deploy v2
2. v2 pod added
3. once v2 is ready, v1 dies

   ~repeat 2 to 3~

4. all replaced

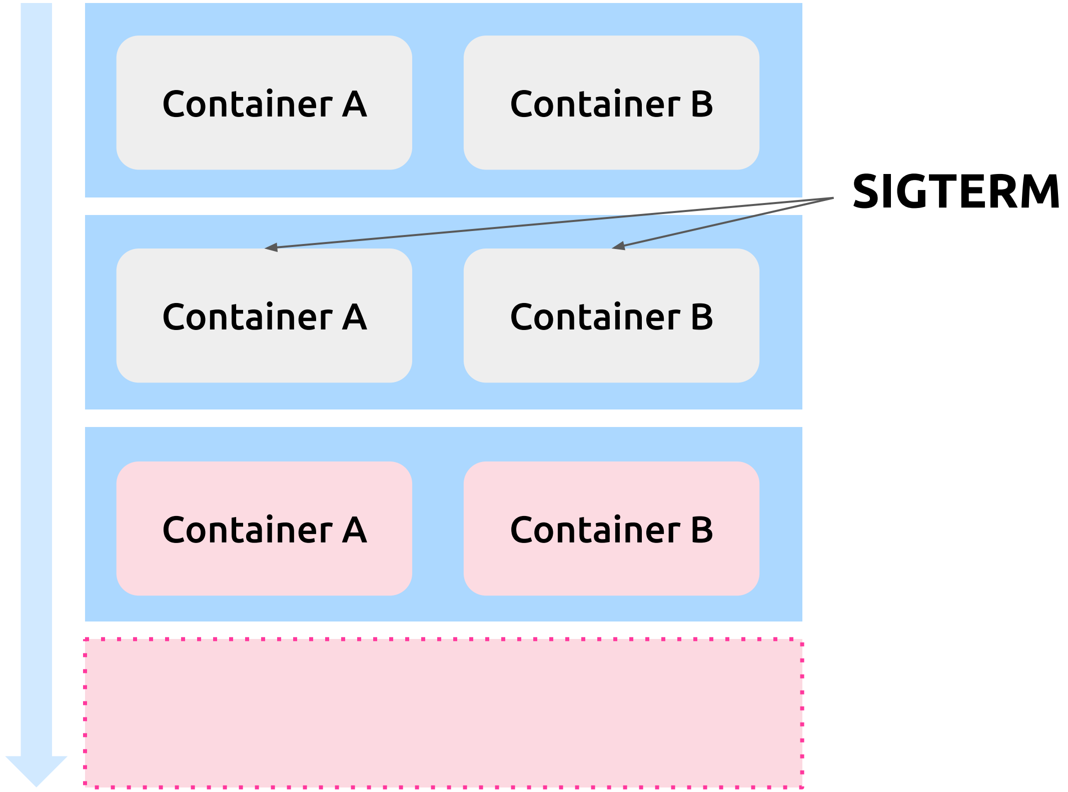


Rails architecture on k8s



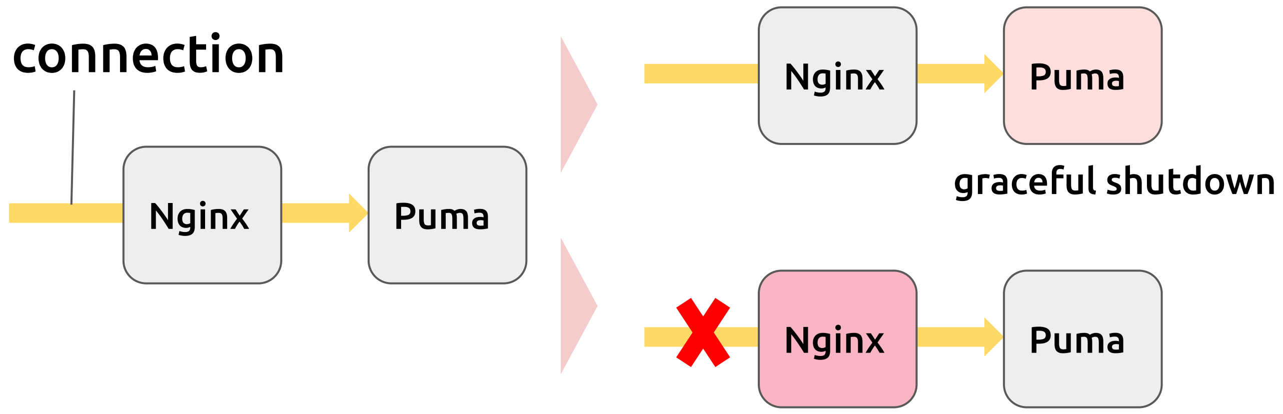
k8s pod lifecycle

1. Processes running
2. Receives SIGTERM
3. Each process dies
4. Pod dies



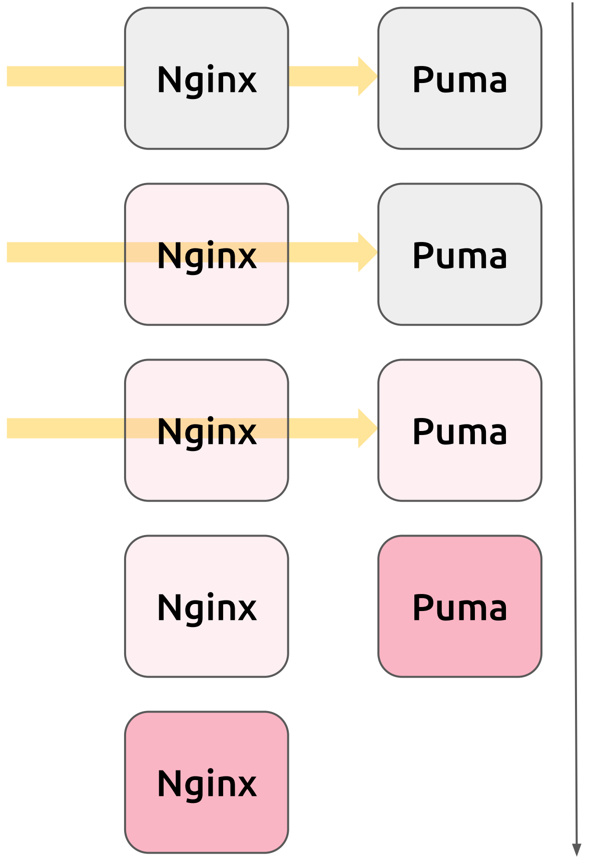
Graceful Shutdown Problem

- By default, shutdown is not ordered
    - Puma start Graceful Shutdown
    - Nginx immediately dies
    - connection badly closed X(



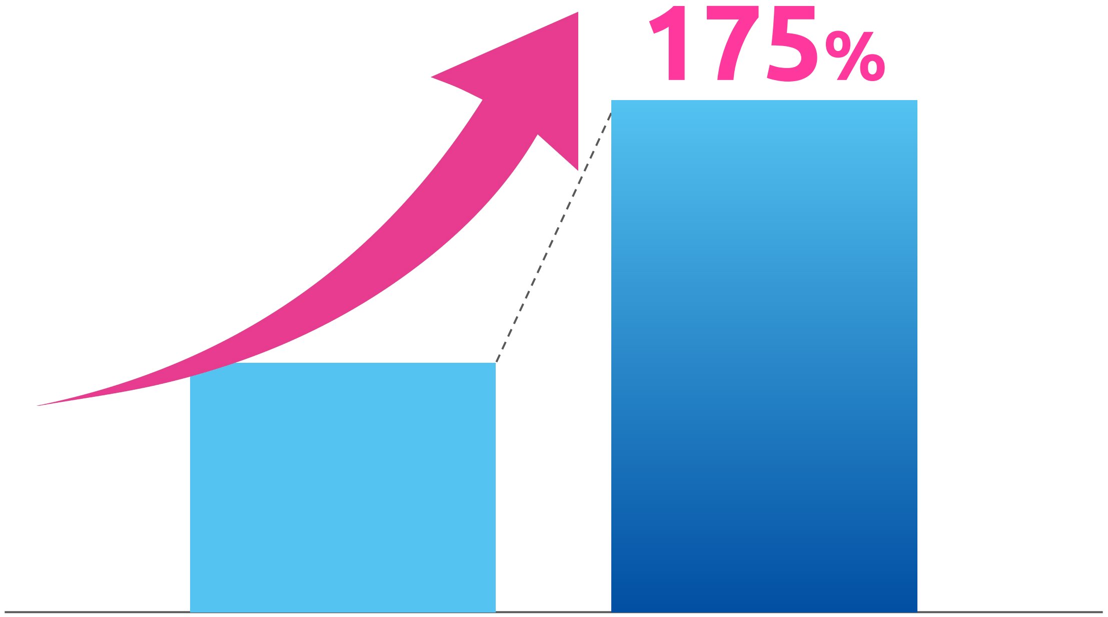
Graceful Shutdown Problem: How To Order

1. Puma & Nginx postpones SIGTERM
   a. preStop
2. Start GS in Nginx
   a. till wait puma process ends
3. Start GS in Puma
   a. connections stay
4. Requests gone, receives SIGTERM
   a. Nginx starts shutdown
5. Pod dies



Achievements

More Productivity



175% productivity since previous architecture

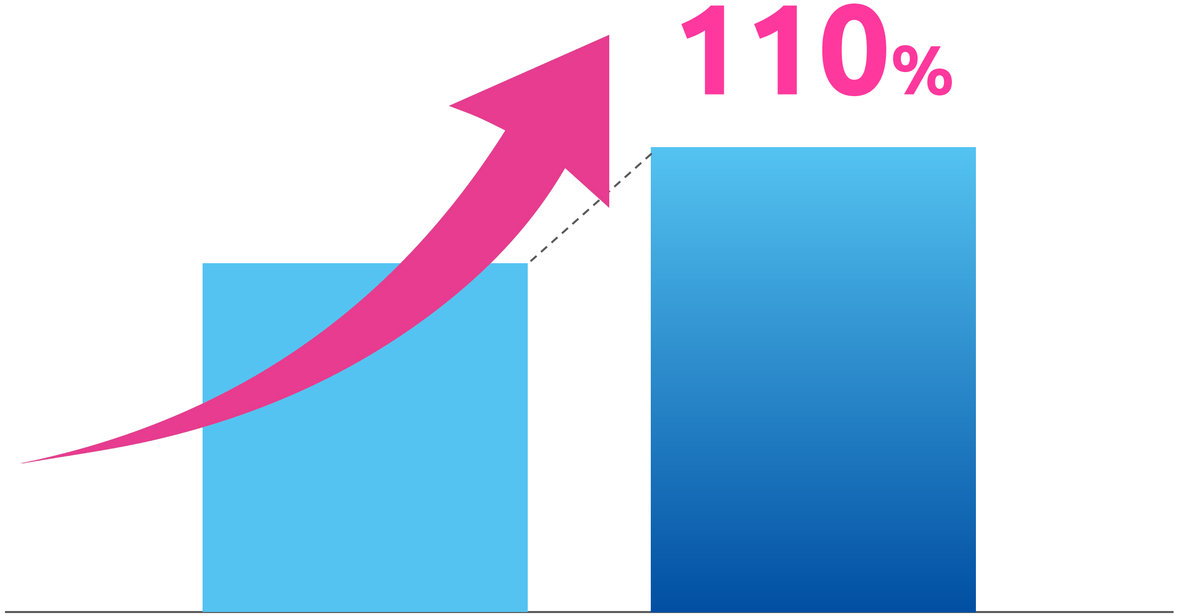
Fast Velocity

250%

velocity since previous architecture

30 releases in first 3 months

CVR rift



by UX improvements

Wrap up

TL; DR

To focus on users,

Ruby on Rails on Kubernetes on Clouds

fin.