ARCHITECTING FOR OPERATIONS

BORING IS AWESOME!

TWITCH

- Feel free to ask questions in the chat at anytime!
- Giving back feedback through the chat helps me read the audience.
- Please respond to each other in chat as well.
- Big thanks to the volunteer student moderators:
  - Wolgo
  - CptWesley
- We are going to have a break at 1630±

AUDIENCE

- Are you working in the industry?
- Are you operating infrastructure?
- What do you expect from this lecture?

MINDSET

COLLABORATION

SERVING THE CUSTOMER

IMPACT

STEFFAN NORBERHUIS

- Freelance Cloud & DevOps Consultant
- Twitter: @SNorberhuis
- steffan@norberhuis.nl
- Feel free to contact me!

OVERVIEW

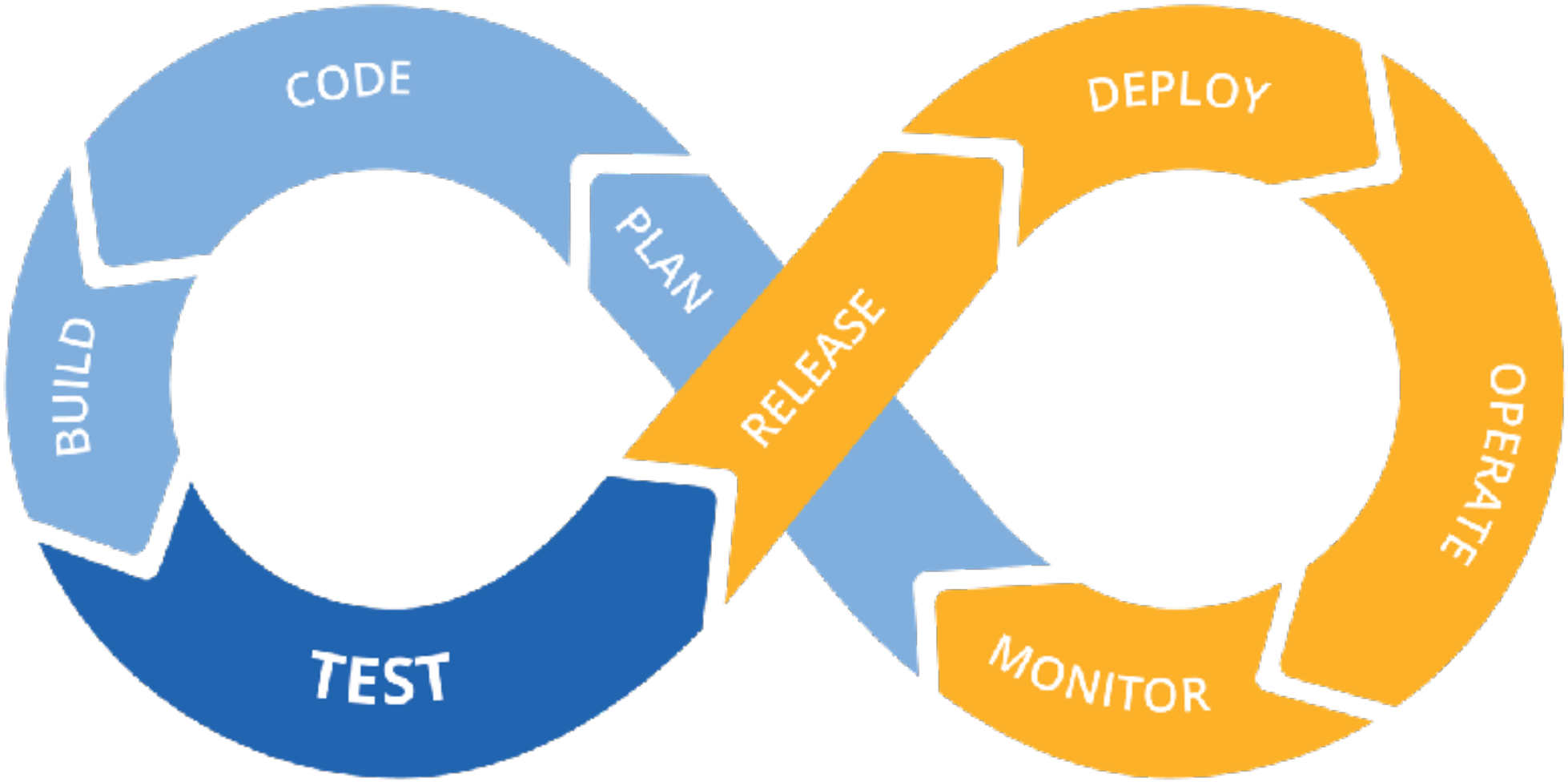
- Disruption
- Infrastructure as Code
- Failure is inevitable
- Building for Failure

ARCHITECTING FOR OPERATIONS

DISRUPTION

YOU BUILD IT, YOU RUN IT

DEVOPS OWNERSHIP



CLOUD

CLOUD

- Operate technology without owning technology
- Agility with no planning
- Focus on your business

Source: What is Cloud Computing by AWS https://www.twitch.tv/videos/477810350?

ARCHITECTING FOR OPERATIONS

INFRASTRUCTURE AS CODE

BENEFITS

- Automation
- Version control
- Code Review
- Testing
- Documentation
- Reuse

Source: 5 Lessons Learned From Writing Over 300,000 Lines of Infrastructure Code by Yevgeniy Brickman
https://www.youtube.com/watch?v=RTEgE2Icyk4 https://www.youtube.com/watch?v=RTEgE2Icyk4

INFRASTRUCTURE AS CODE

AWS CloudFormation

HashiCorp Terraform

Troposphere

Pulumi

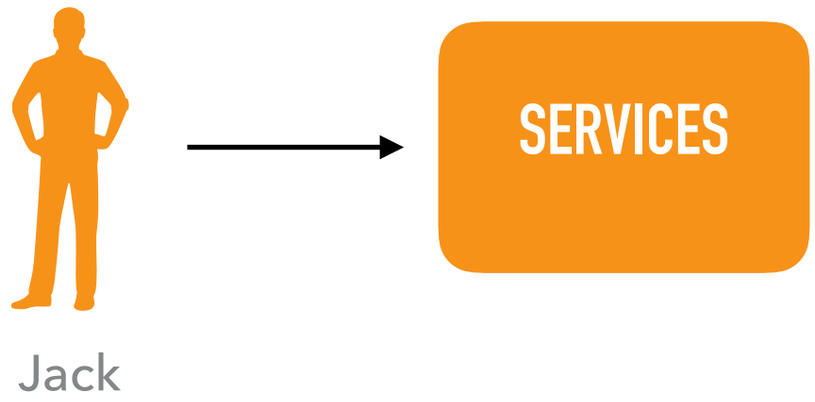
Azure Resource Manager

AWS Cloud Development Kit

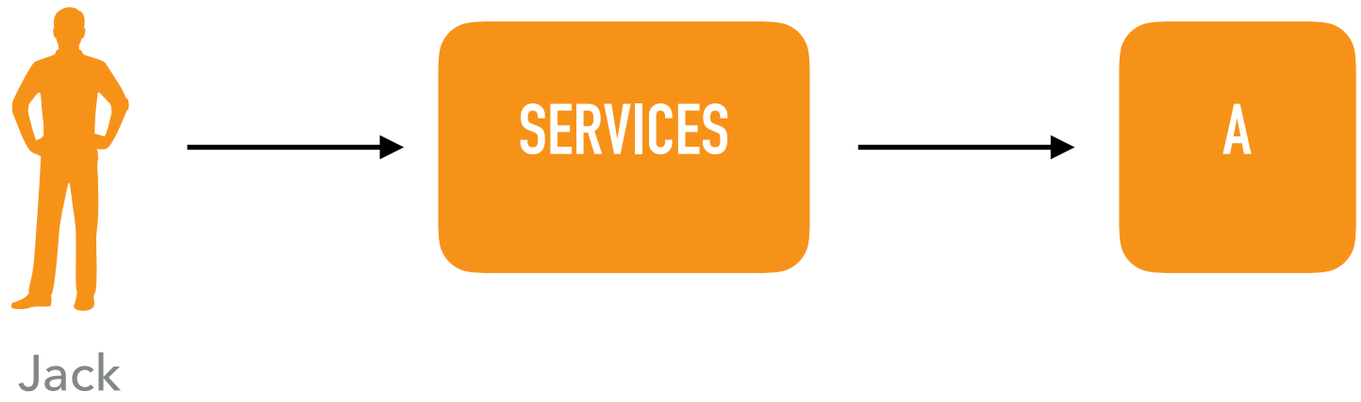
JSON / YAML | Declarative | DOM | Componentized

Source: AWS CDK by AWS re:Invent https://www.youtube.com/watch?v=Lh-kVC2r2AU

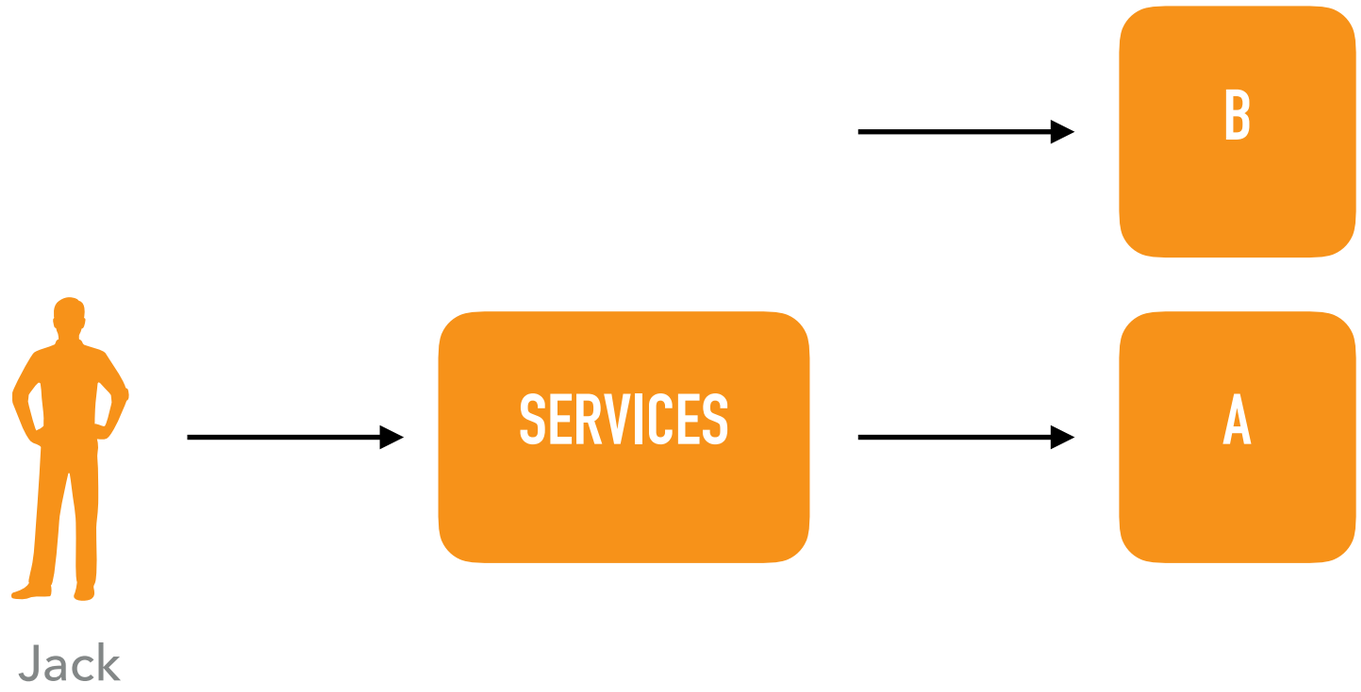
SLOW FEEDBACK LOOP



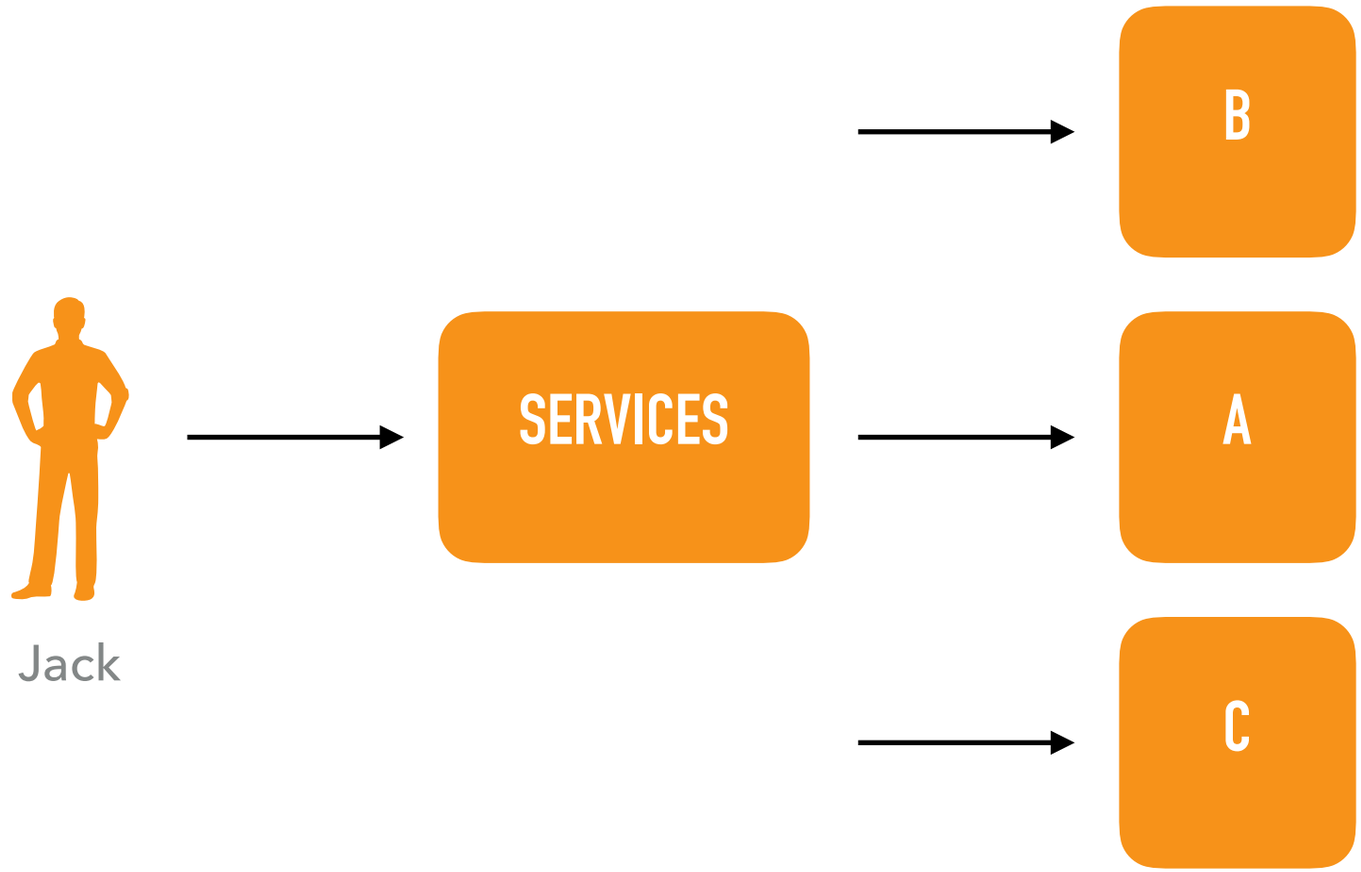
SLOW FEEDBACK LOOP



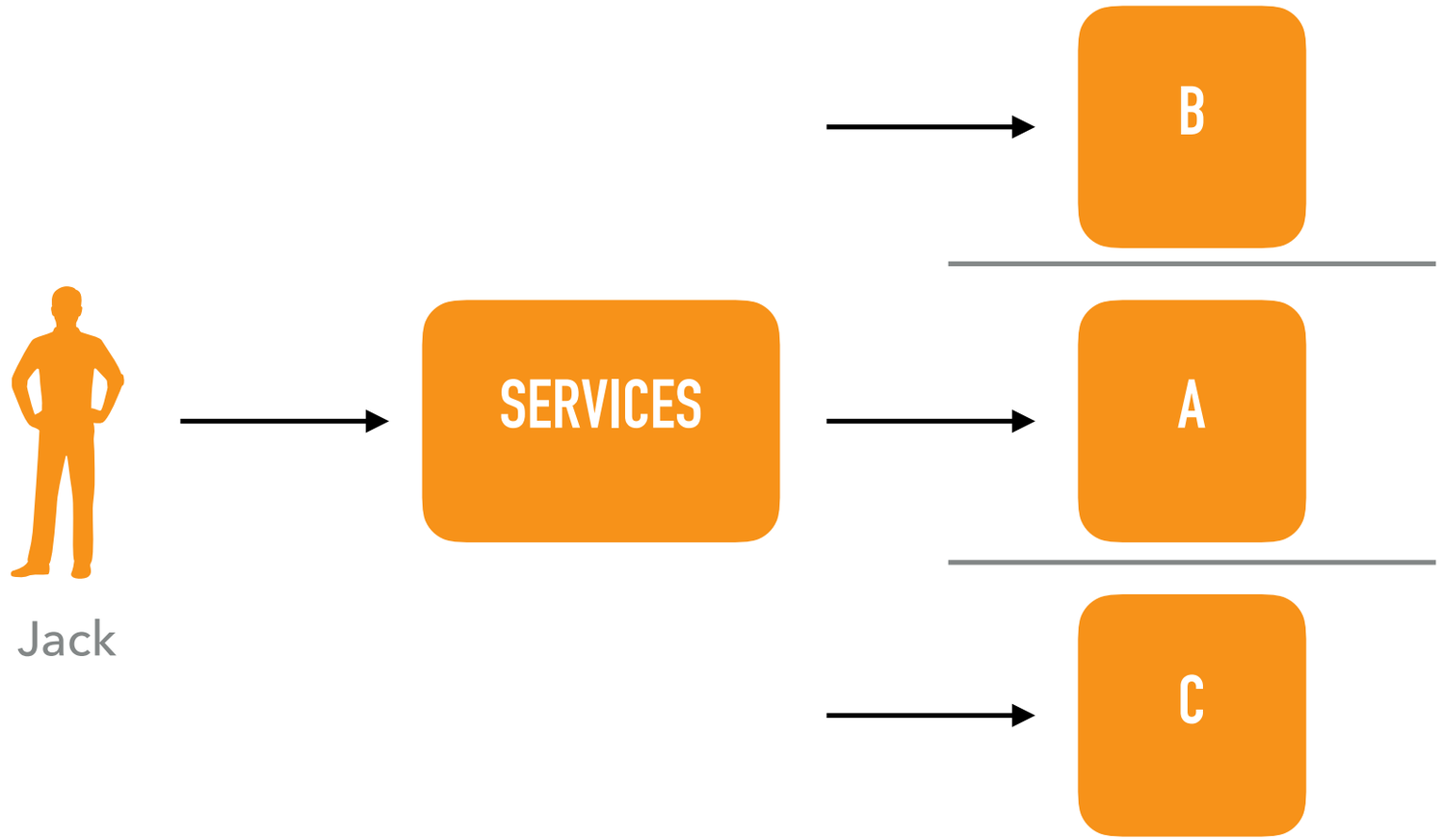
SLOW FEEDBACK LOOP



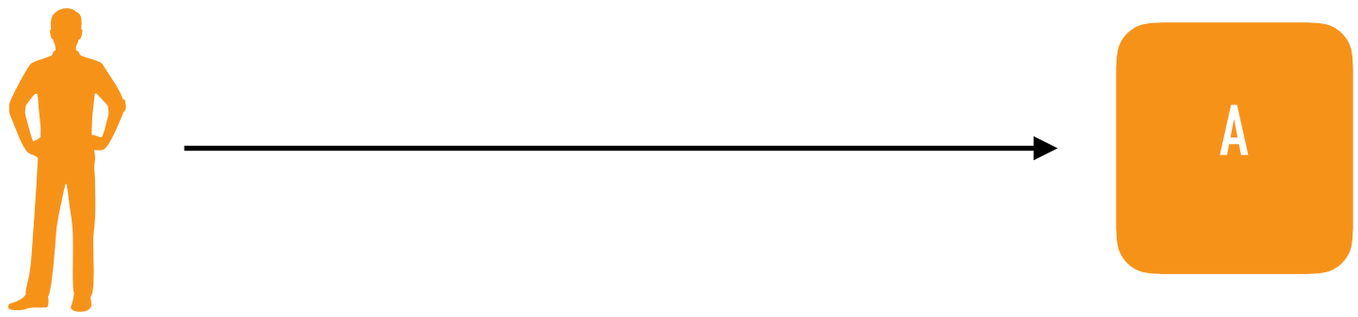
SLOW FEEDBACK LOOP



SLOW FEEDBACK LOOP

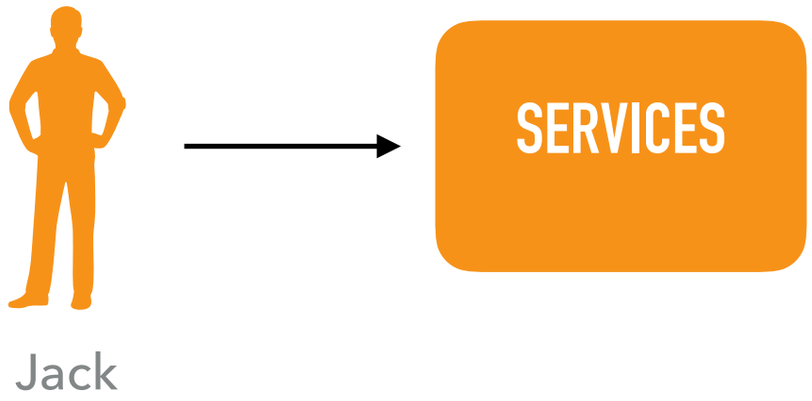


SLOW FEEDBACK LOOP



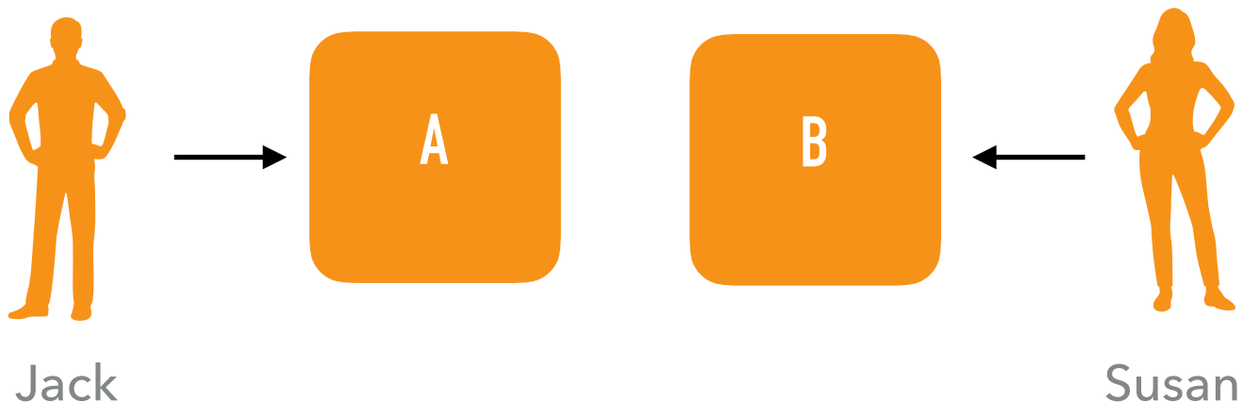
Jack

COLLISIONS DURING DEVELOPMENT



COLLISIONS DURING DEVELOPMENT

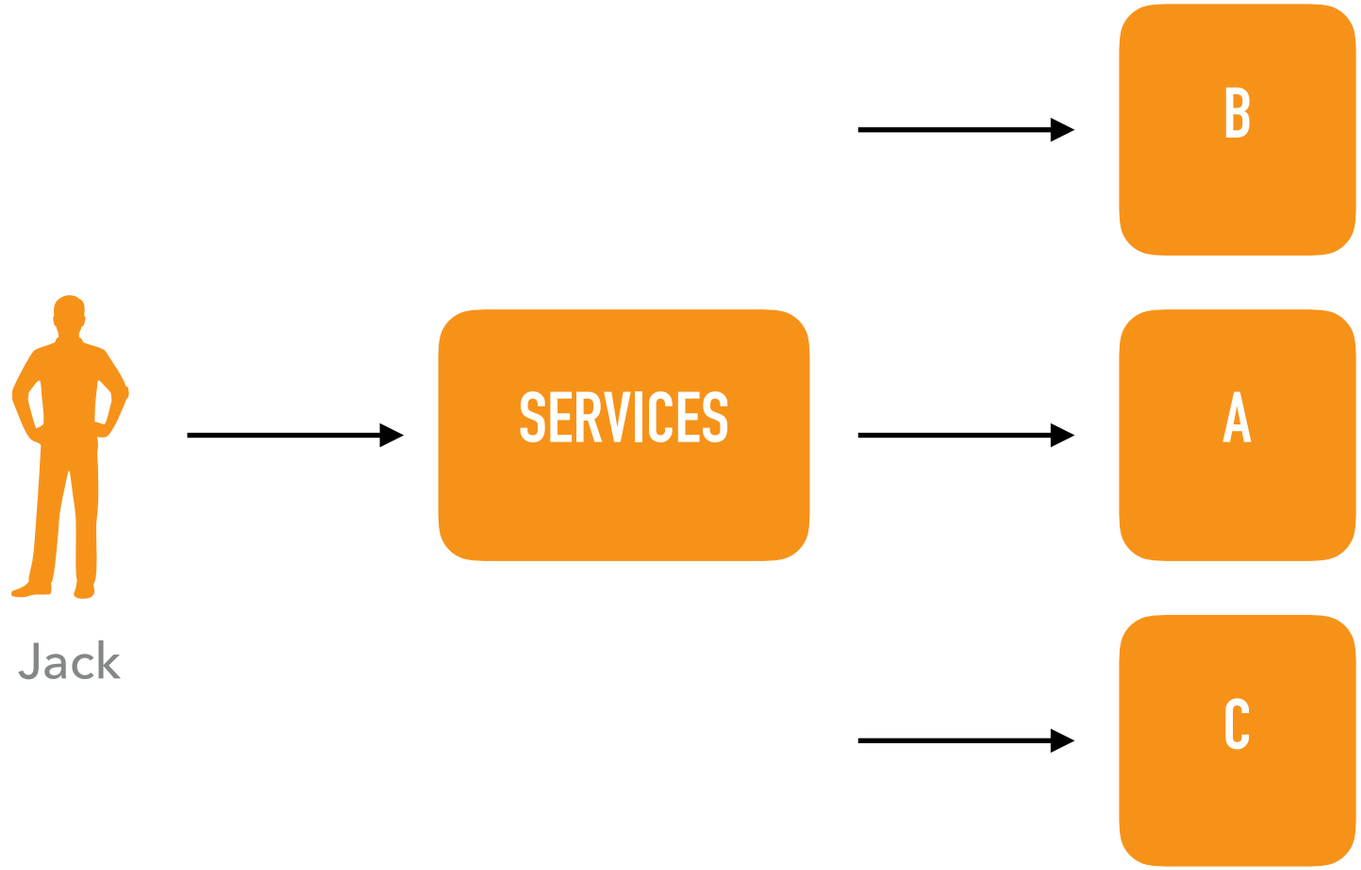
COLLISIONS DURING DEVELOPMENT



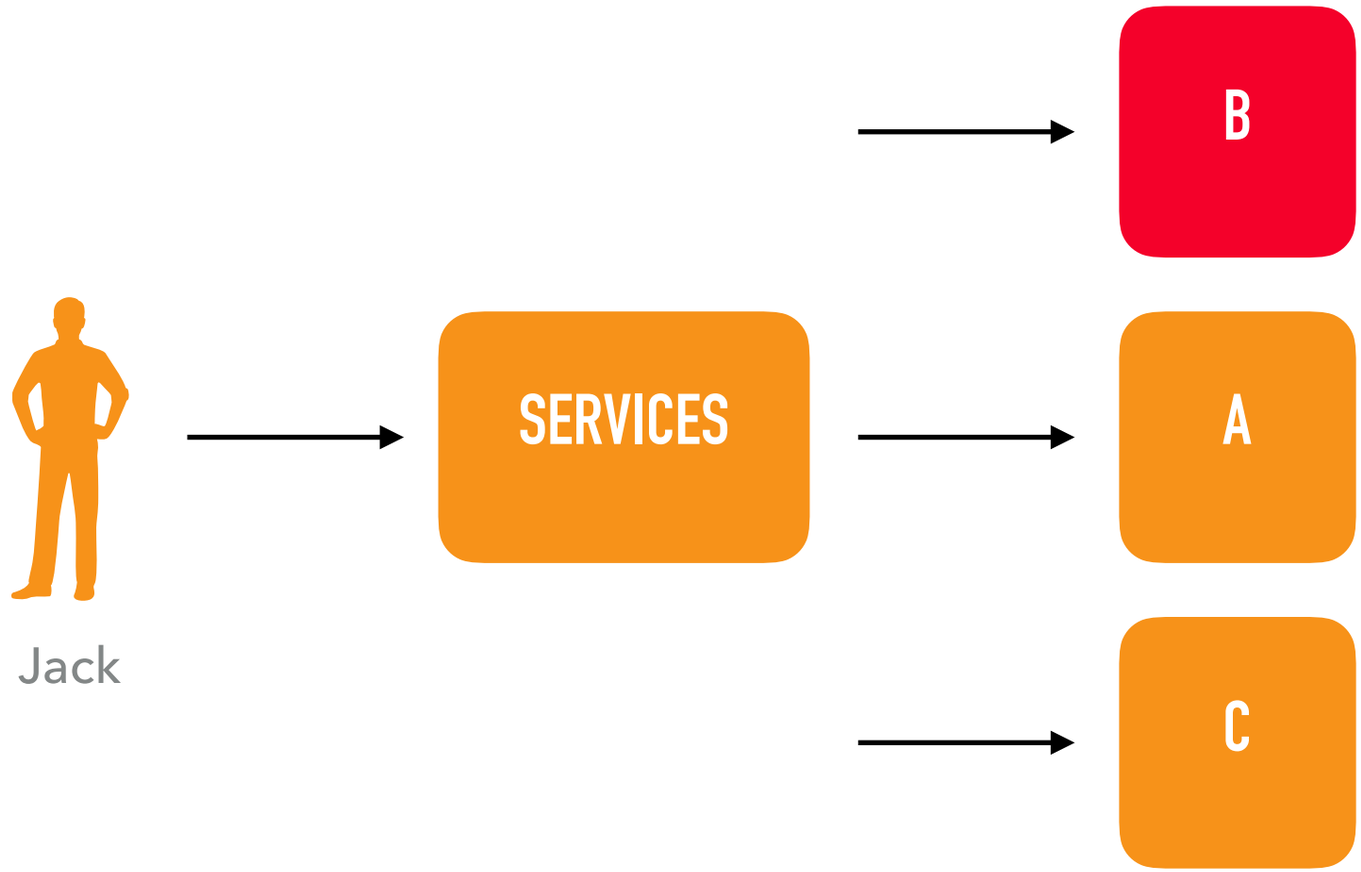
COLLISIONS DURING DEVELOPMENT

COLLISIONS DURING DEVELOPMENT

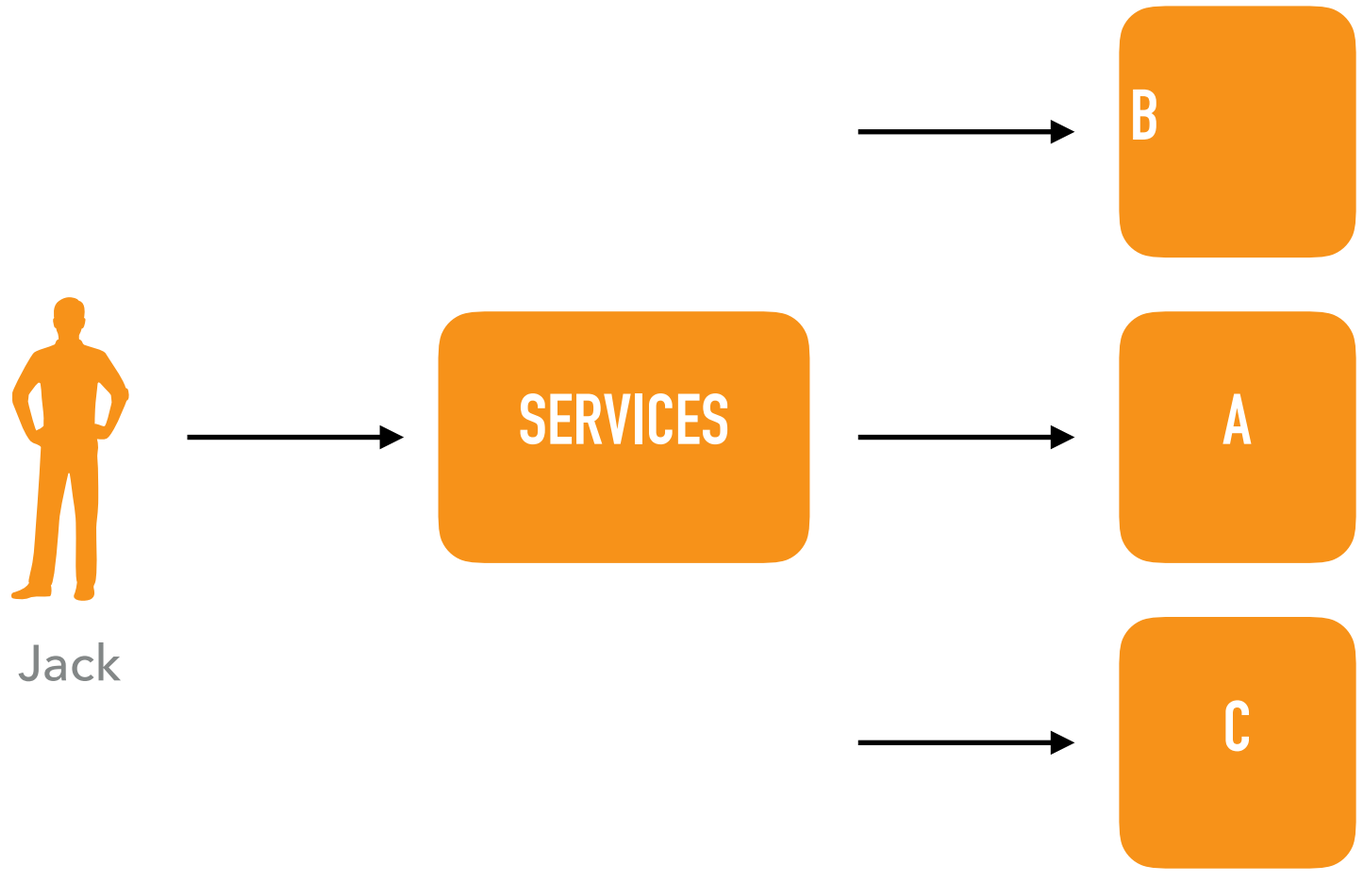
PROBLEMS



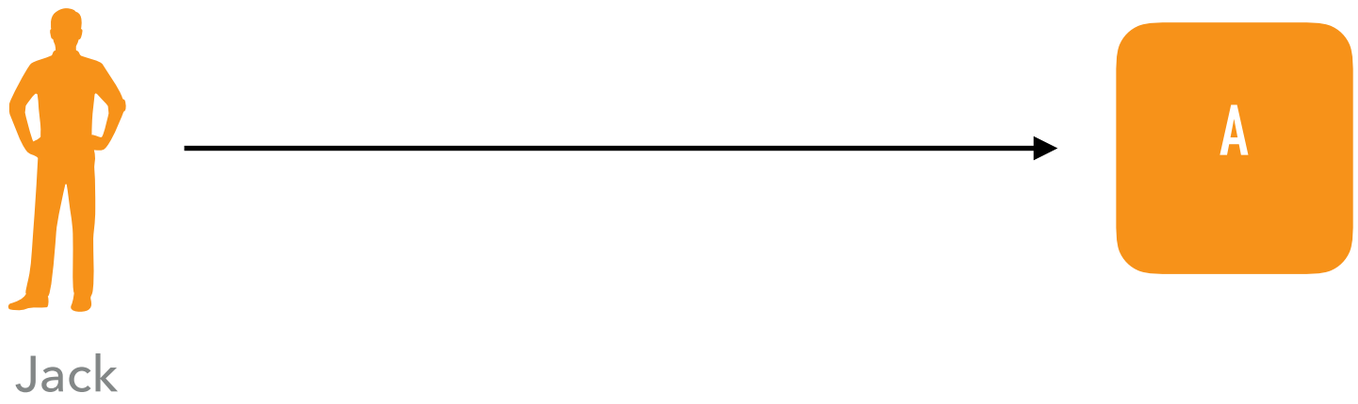
PROBLEMS: BUGS



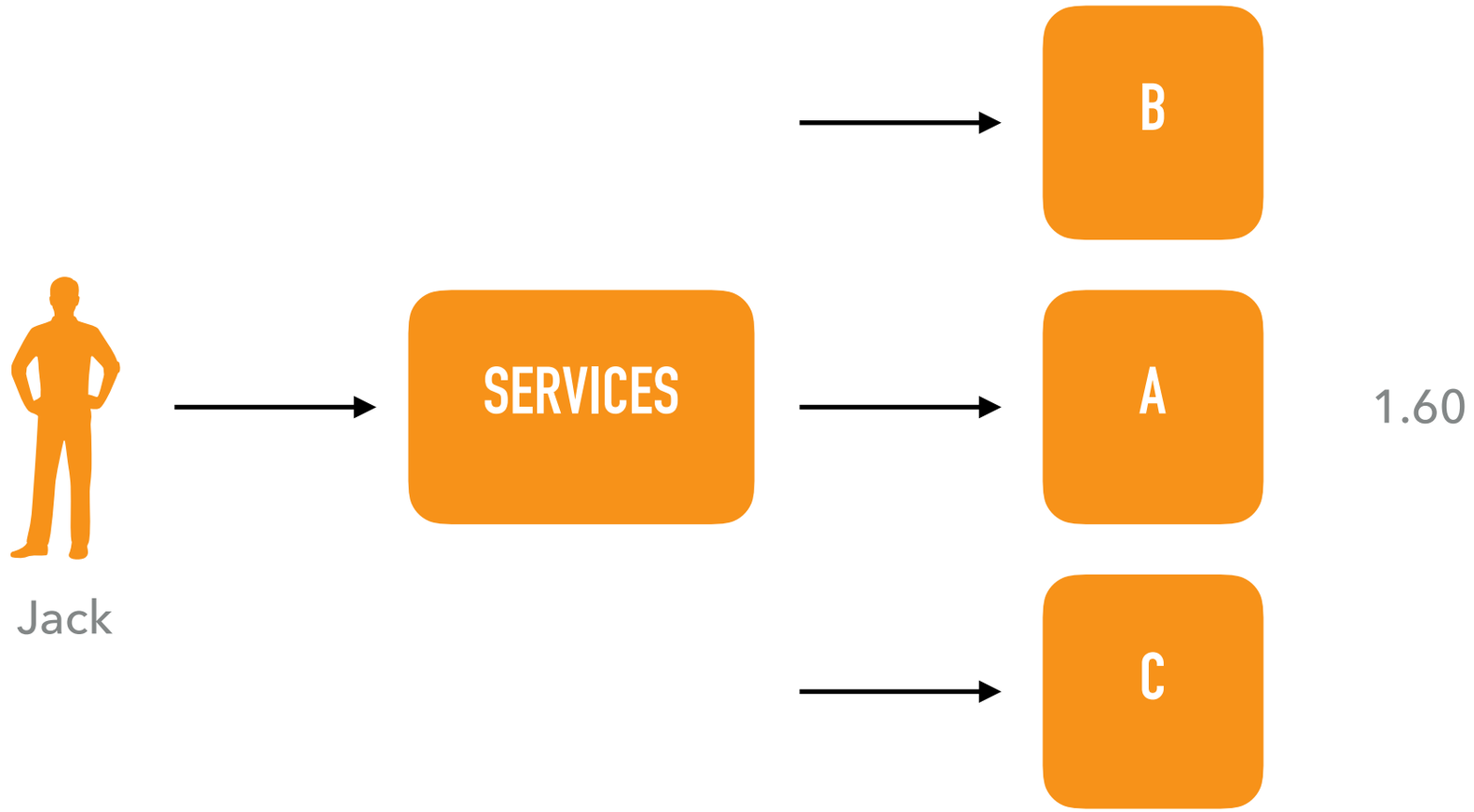
PROBLEMS: DRIFT



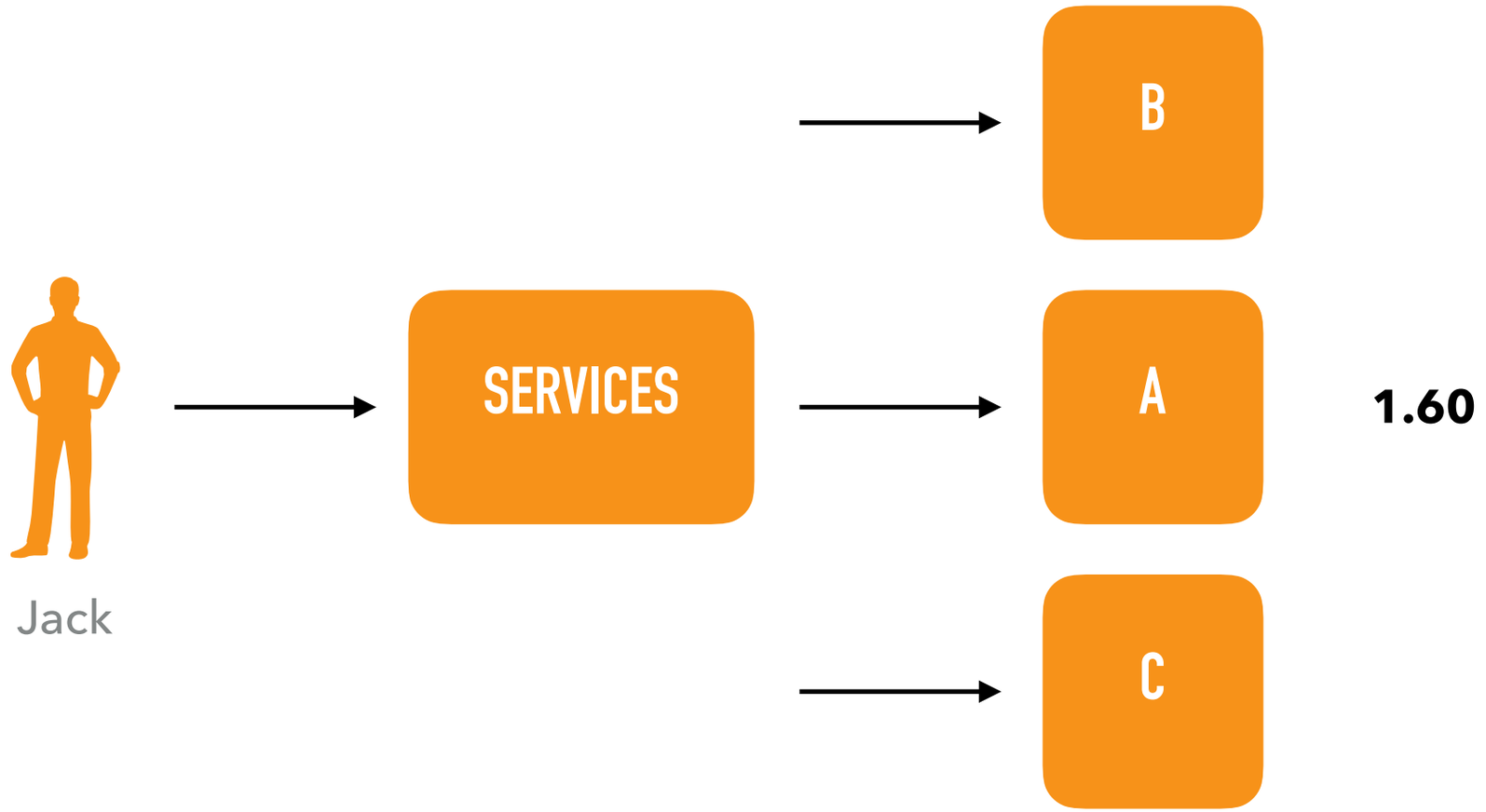
PROBLEMS



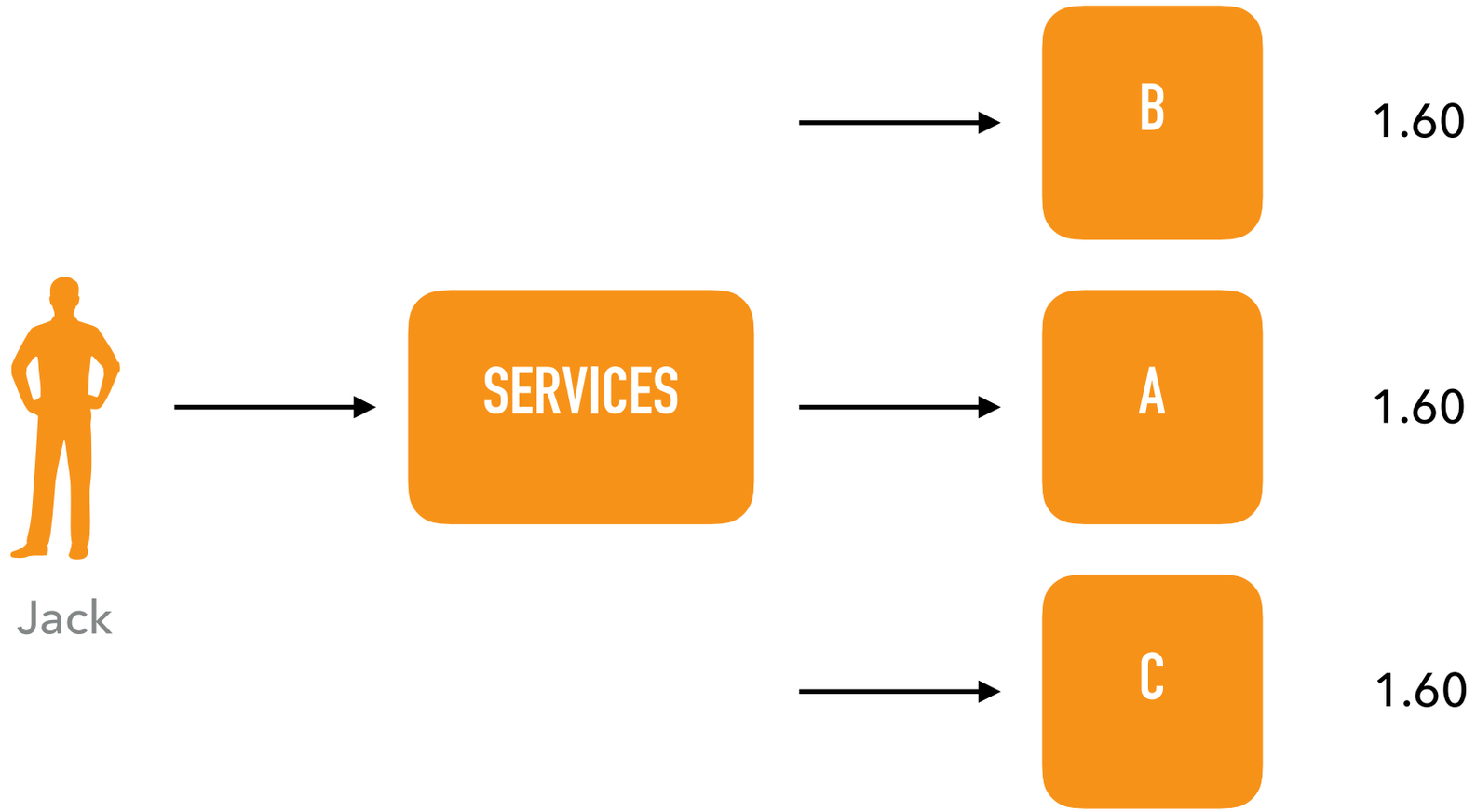
TAGGING/BRANCH DEADLOCKS



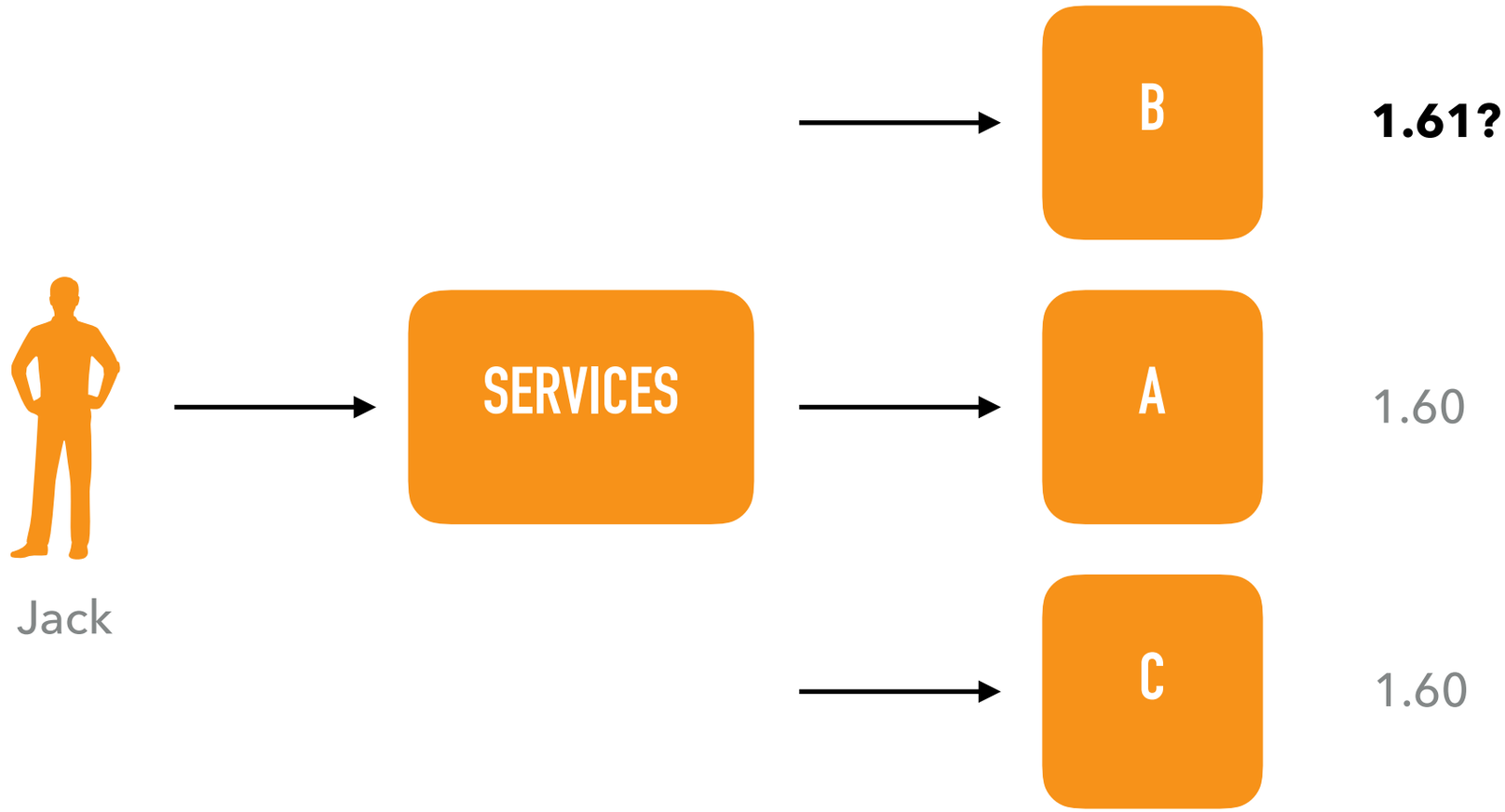
TAGGING DEADLOCKS



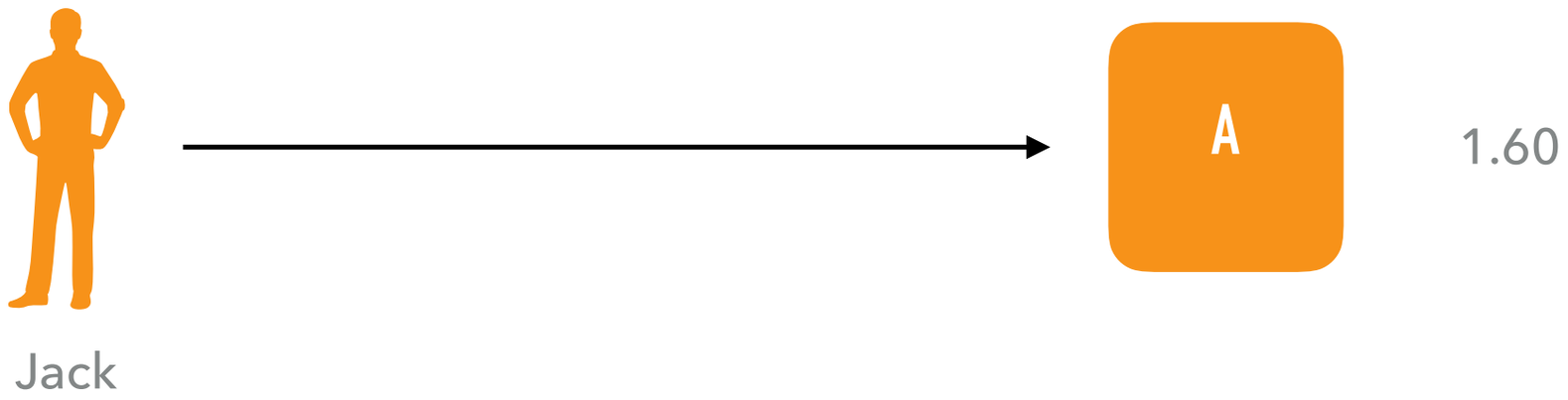
TAGGING DEADLOCKS



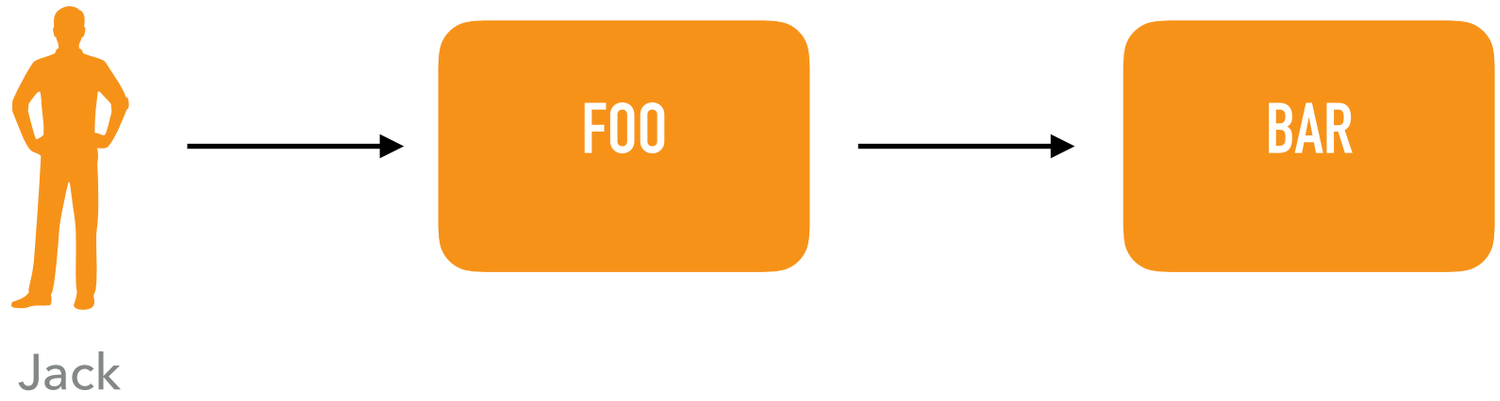
TAGGING DEADLOCKS



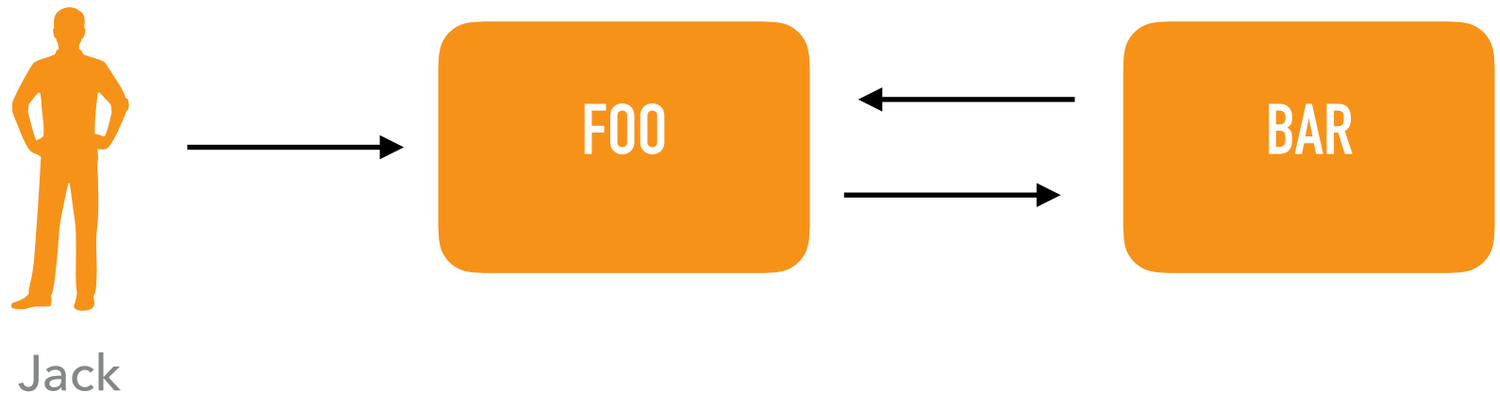
TAGGING DEADLOCKS



CYCLOMATIC DEPENDENCY



CYCLOMATIC DEPENDENCY



CHALLENGES

- Feedback speed
- Parallel development
- Complexity
- Different lifecycles
- Different teams

Source: Happy Terraforming! By Armin Coralic: https://www.youtube.com/watch?v=G06j6HLWyYo

GUIDELINES

- Less frequent changes, higher risk, in lower layers
- Small blocks
- No cyclomatic dependencies
- Decouple independent services
- Only deploy pipelines manually

ARCHITECTURE

DOMAIN

CONFIG | DATA | APPLICATION | SECRETS

INFRA

VPC | NETWORK

GLOBAL

ALERTING | CONFIG

BOOTSTRAP

ACCOUNTS | ROLES | SECURITY

ARCHITECTURE

DOMAIN

CONFIG | DATA | APPLICATION | SECRETS

INFRA

VPC | NETWORK

GLOBAL

ALERTING | CONFIG

BOOTSTRAP

ACCOUNTS | ROLES | SECURITY

ARCHITECTURE

DOMAIN

CONFIG

DATA

APPLICATION

SECRETS

INFRA

VPC

NETWORK

GLOBAL

ALERTING

CONFIG

BOOTSTRAP

ACCOUNTS

ROLES

SECURITY

ARCHITECTURE

DOMAIN

CONFIG

DATA

APPLICATION

SECRETS

INFRA

VPC

NETWORK

GLOBAL

ALERTING

CONFIG

BOOTSTRAP

ACCOUNTS

ROLES

SECURITY

ARCHITECTURE

DOMAIN

CONFIG

DATA

APPLICATION

SECRETS

INFRA

VPC

NETWORK

GLOBAL

ALERTING

CONFIG

BOOTSTRAP

ACCOUNTS

ROLES

SECURITY

ARCHITECTURE

DOMAIN

CONFIG | DATA | APPLICATION | SECRETS

INFRA

VPC | NETWORK

GLOBAL

ALERTING | CONFIG

BOOTSTRAP

ACCOUNTS | ROLES | SECURITY

NAMING STANDARDISATION

- Environment
- Application
- Component

- Examples:
  - /prod/billing/foo
  - /dev-susan/billing/foo
  - staging-billing-foo

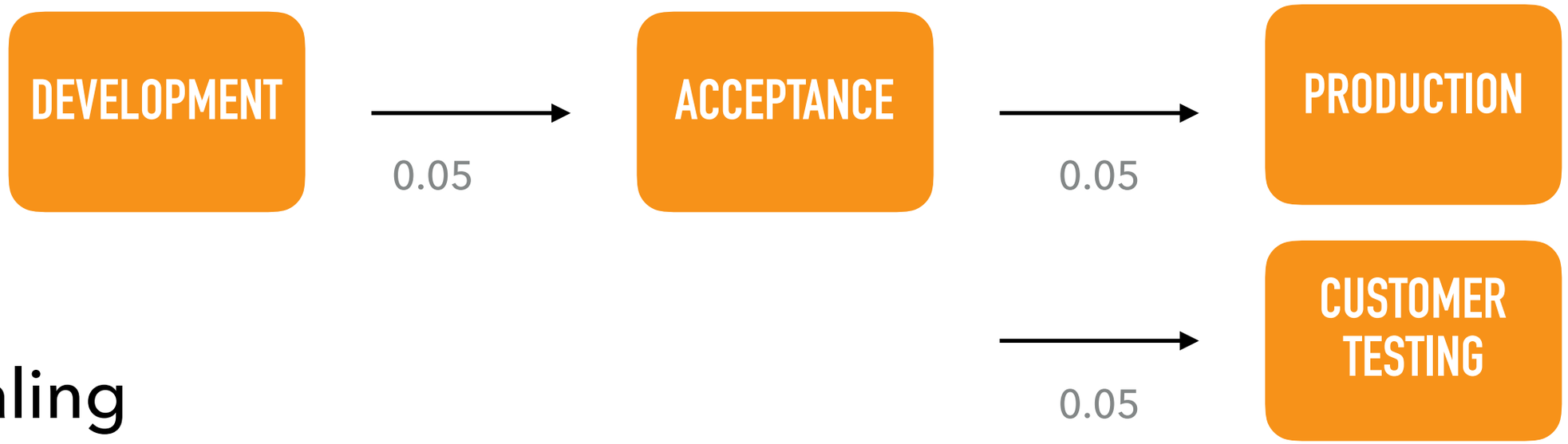
CODE TRACEABILITY

- Tag:
  - github.com/org/teamA/billing-infrastructure/stackA
- Naming:
  - Billing-application-foo -> GitHub.com/org/billing/infrastructure/src/application/foo

IDENTICAL ENVIRONMENTS

- Scaling

IDENTICAL ENVIRONMENTS



- Scaling
- Multiple environments

IDENTICAL ENVIRONMENTS

- Scaling
- Multiple environments
- Acceptance tests everything

OPEN SOURCE

- Terraform: https://github.com/terraform-community-modules
- AWS CDK: https://cdkpatterns.com/
- AWS CloudFormation: https://aws.amazon.com/quickstart/?
- Gruntwork*: https://www.gruntwork.io/

PATH OF ENLIGHTENMENT

DEVOPS METRICS

LEAD TIME

DEPLOYS

CHANGE FAILURE RATE

MEAN TIME TO RECOVERY

Source: Accelerate!

DEVOPS METRICS

LEAD TIME

DEPLOYS

CHANGE FAILURE RATE

MEAN TIME TO RECOVERY

Source: Accelerate!

DEVOPS METRICS

LEAD TIME

DEPLOYS

CHANGE FAILURE RATE

MEAN TIME TO RECOVERY

Source: Accelerate!

DEVOPS METRICS

LEAD TIME

DEPLOYS

CHANGE FAILURE RATE

MEAN TIME TO RECOVERY

Source: Accelerate!

TEST DRIVEN DEVELOPMENT

ARCHITECTING FOR OPERATIONS

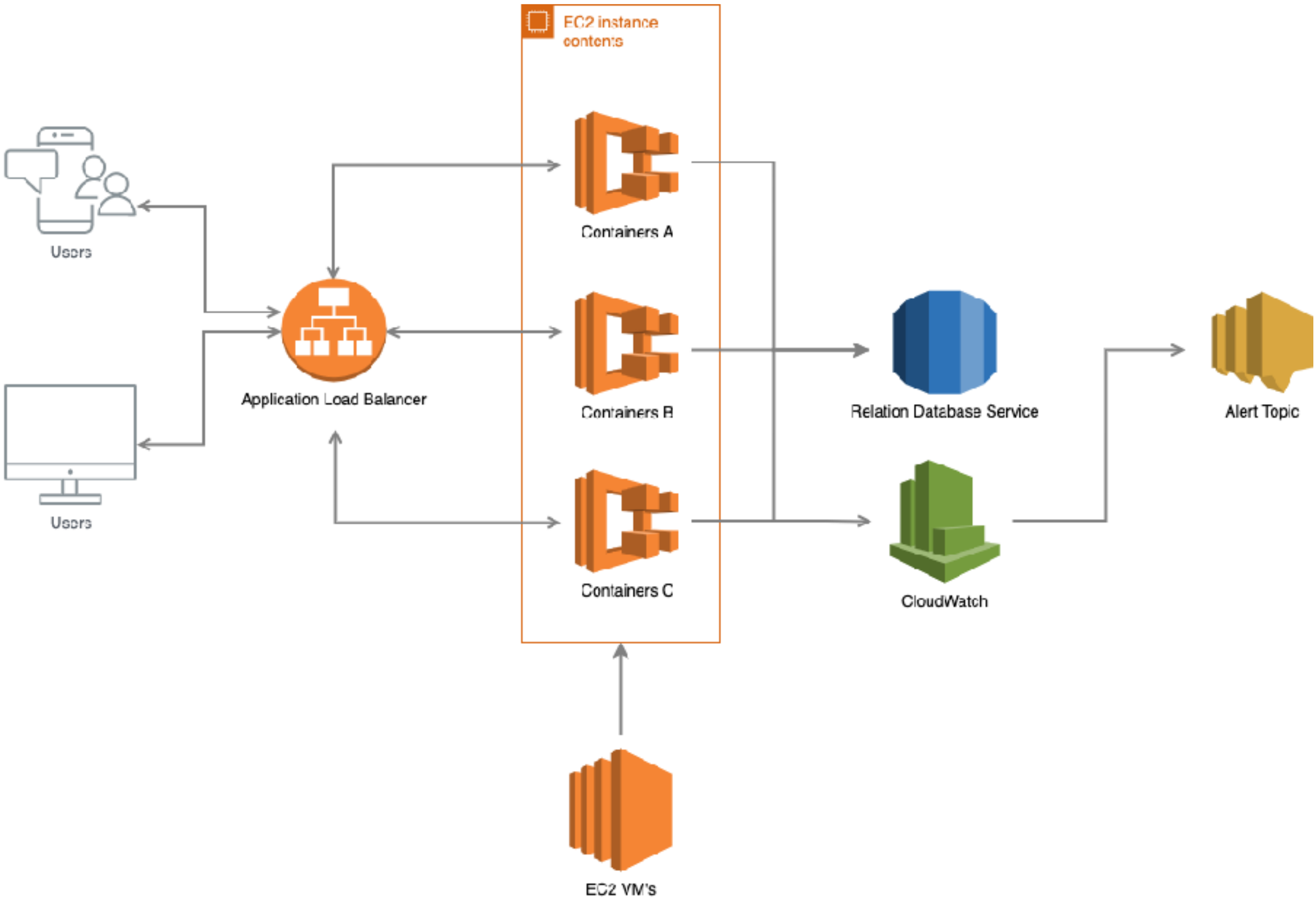
FAILURE IS INEVITABLE

RUNTIME

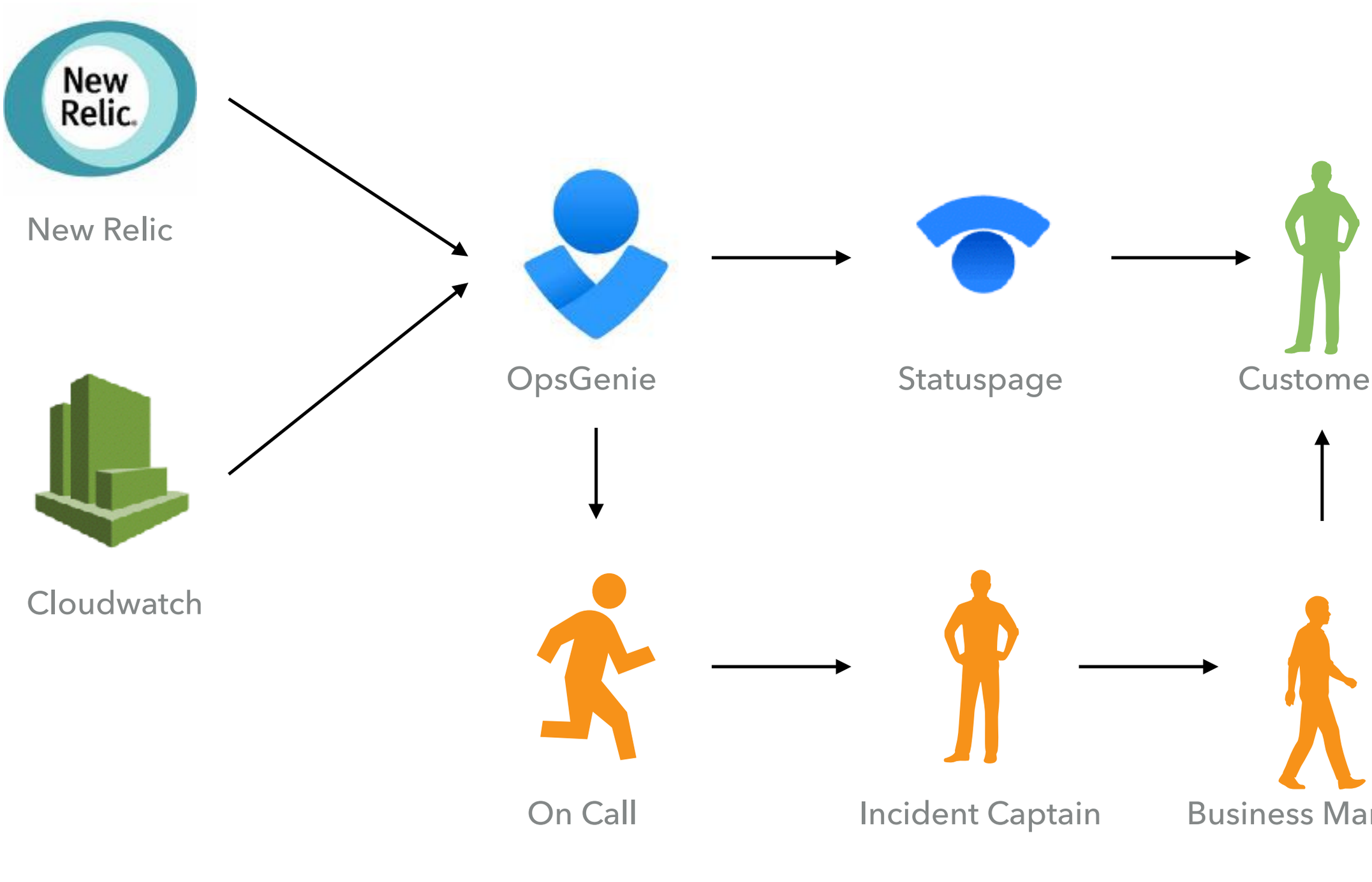
DATA

COMPLEX SYSTEMS

FAILURE IS INEVITABLE



COMMUNICATION



SCIENTIFIC APPROACH

SCIENTIFIC APPROACH

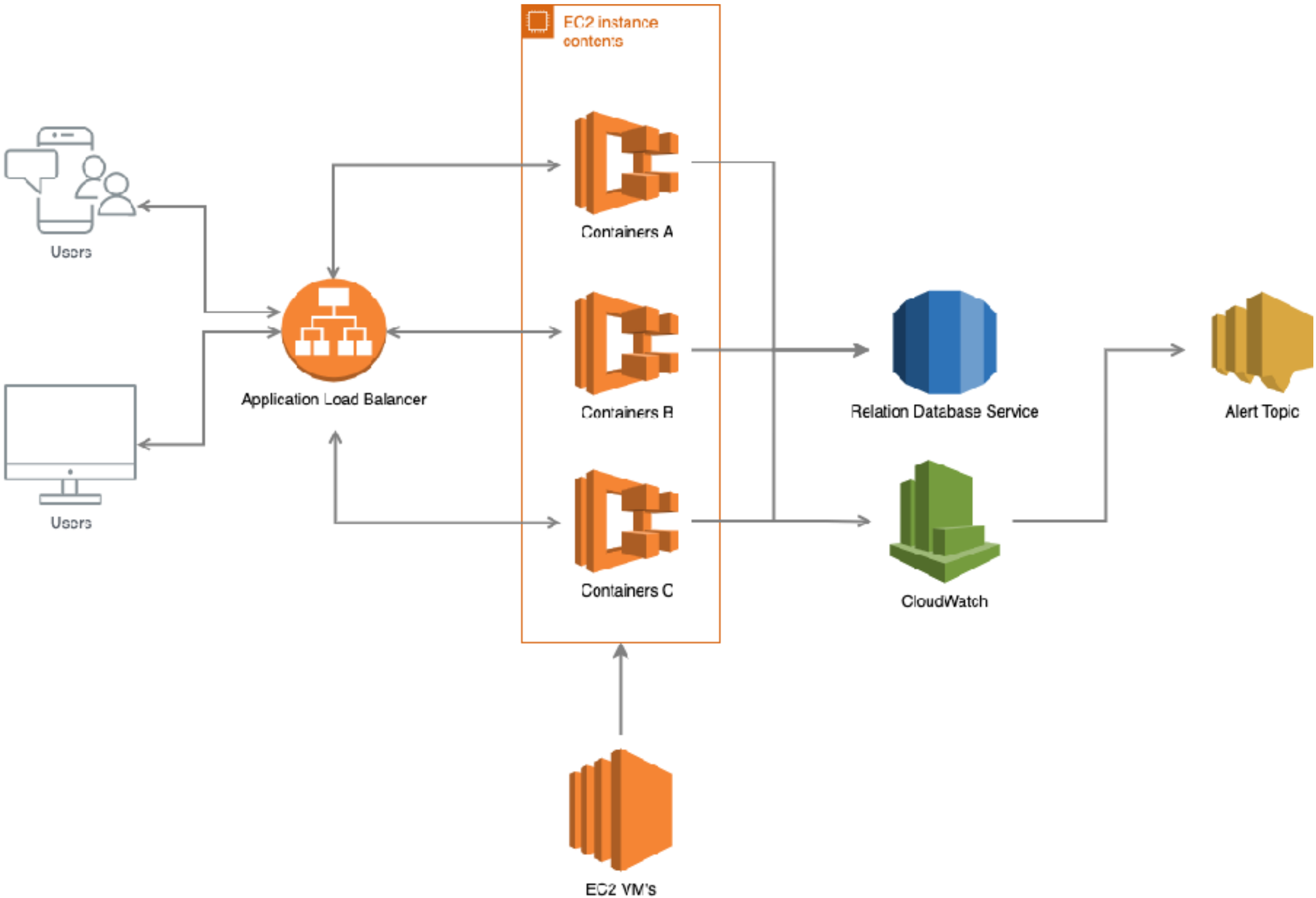
- Describe objectively
- Formulate a hypothesis
- Derive an experiment
- Observe outcomes

LOOK FOR CHANGE

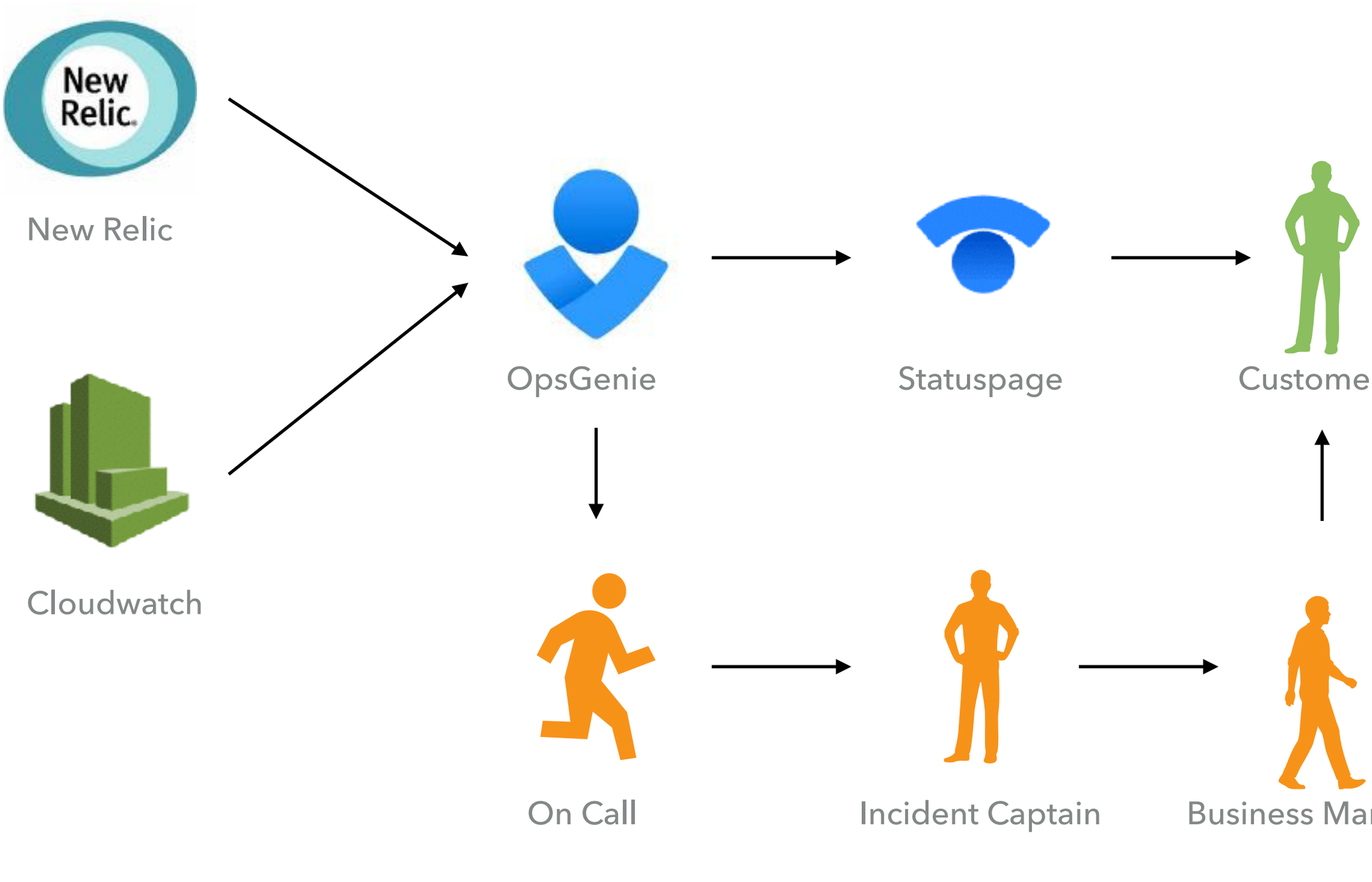
POST MORTEM

TRANSPARENCY

FAILURE IS INEVITABLE



COMMUNICATION



POST MORTEM TEMPLATE

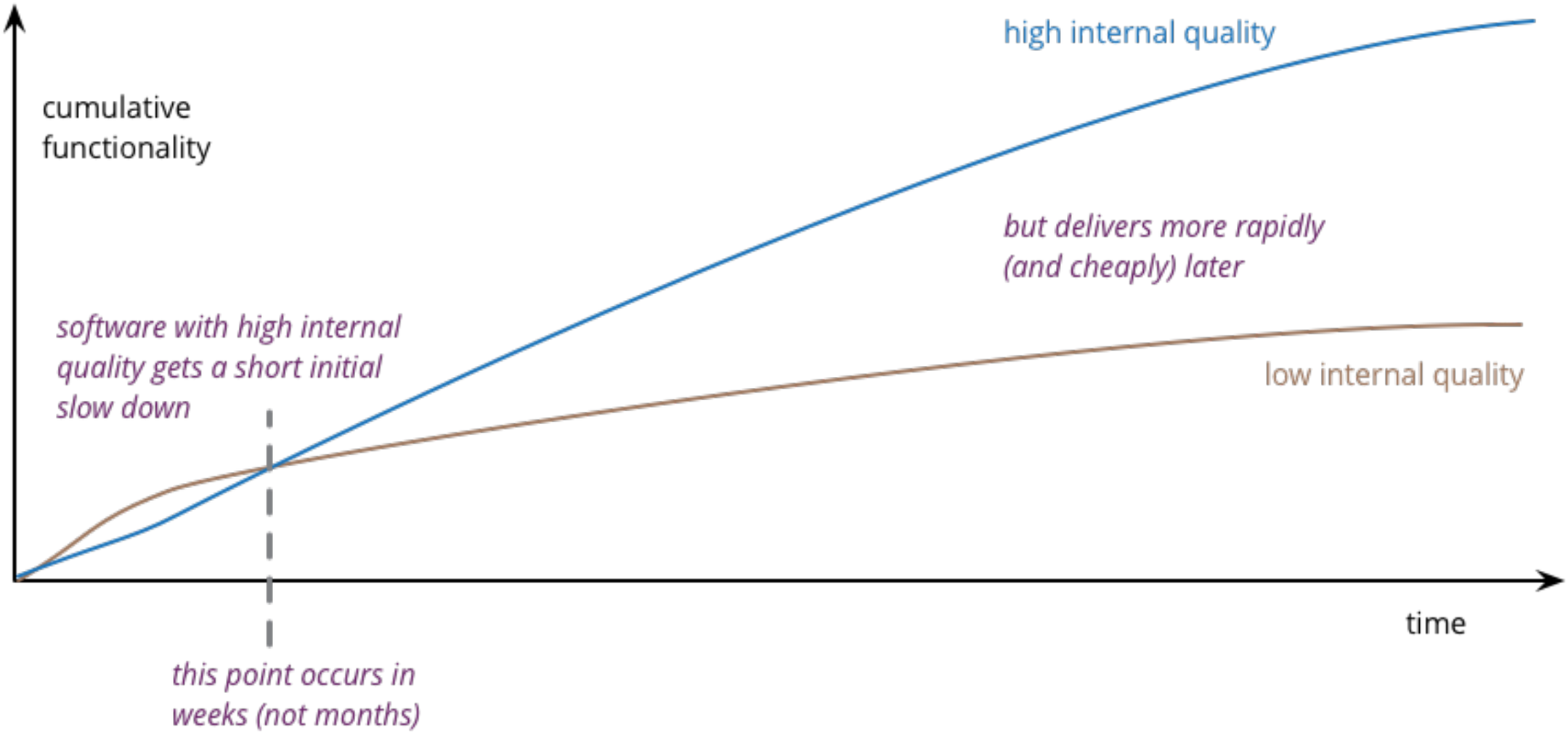
- Timeline: What happened?
  - Impact
  - Resolutions
- Root Cause
- Follow up
  - Public Communication
  - Improvements
    - Organisational
    - Technical

Source:https://response.pagerduty.com/after/post_mortem_process/

UNIQUE FAILURES

NO WORK AROUND

QUALITY



Source: Is High Quality Software Worth the Cost? By Martin Fowler https://www.martinfowler.com/articles/is-quality-worth-cost.html

BROKEN WINDOW THEORY

STAGING

ARCHITECTING FOR OPERATIONS

BUILDING FOR FAILURE


PESSIMIST

DEPLOYMENT STRATEGY

BIG BANG RELEASES

UNSTOPPABLE RELEASES

IF IT HURTS, DO IT MORE OFTEN

BACKWARDS COMPATIBLE

BACKWARDS COMPATIBLE

ALWAYS PUSH TO PRODUCTION

FEATURE TOGGLES

SIMPLE FEATURE TOGGLES

```
function calculate(){

   if( featureToggle("use-new-algorithm") ){

      return newCalculation();

   }else{

      return oldCalculation();

   }

}
```

HOPE IS NOT A STRATEGY

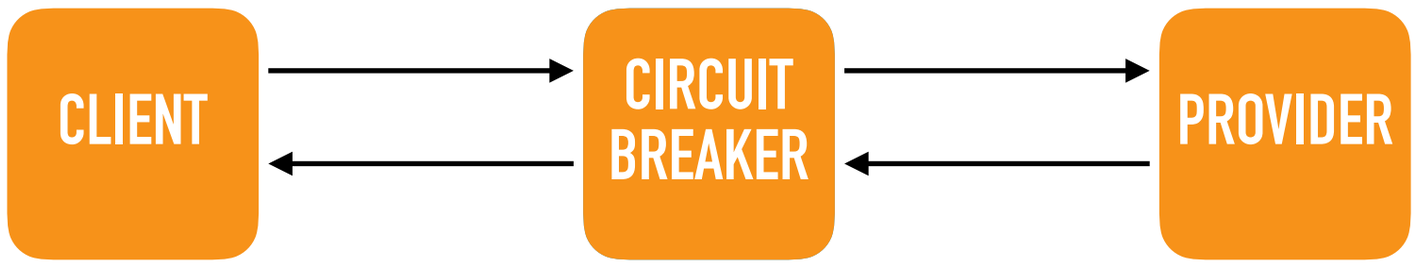
GRACEFUL DEGRATION

GRACEFUL DEGRADATION

- Return less precise data
  - Incomplete data
  - Cached data
  - Preset data
  - No data

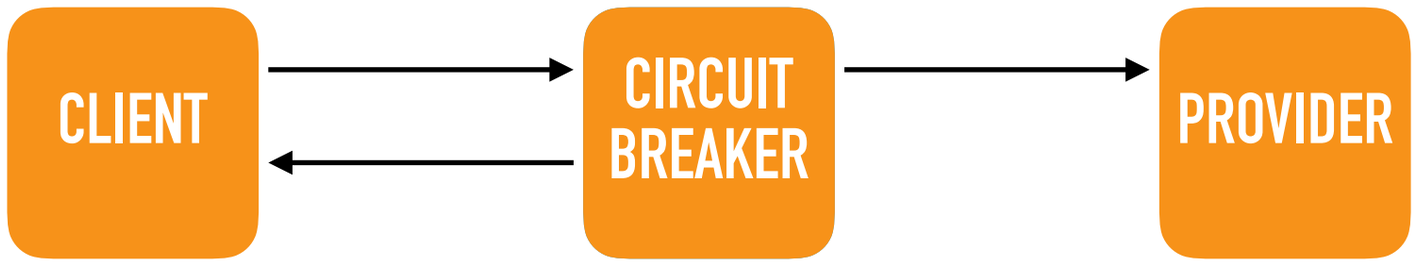
CIRCUIT BREAKER

CIRCUIT BREAKER



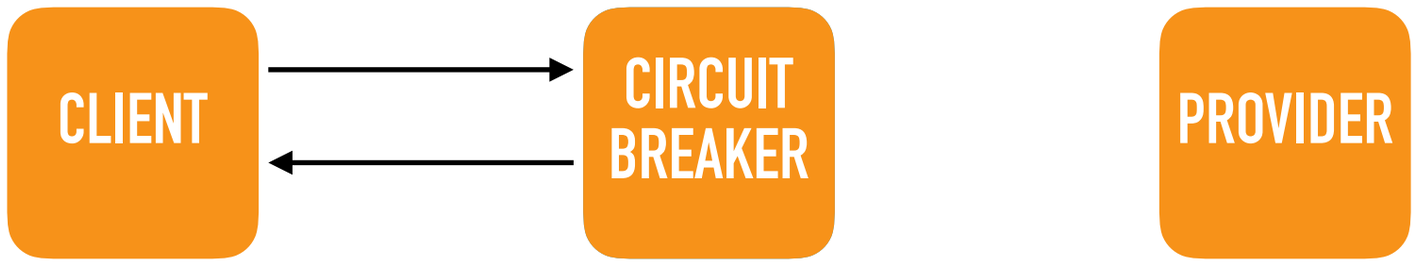
NORMAL

CIRCUIT BREAKER



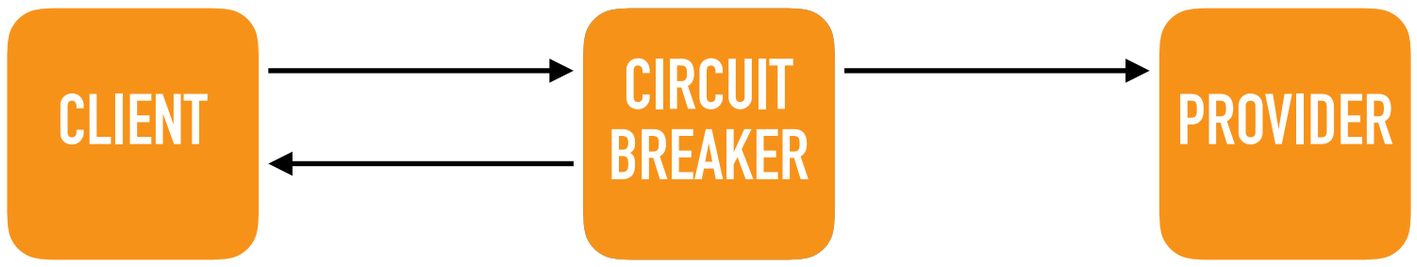
TIMEOUT

CIRCUIT BREAKER



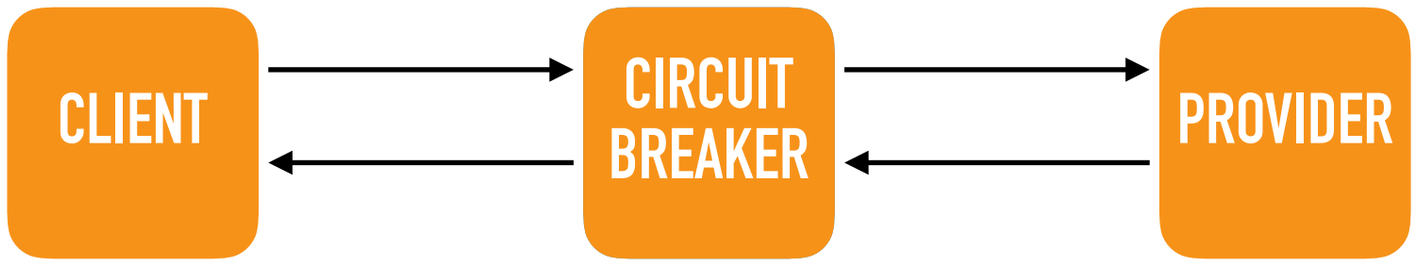
CIRCUIT OPEN

CIRCUIT BREAKER



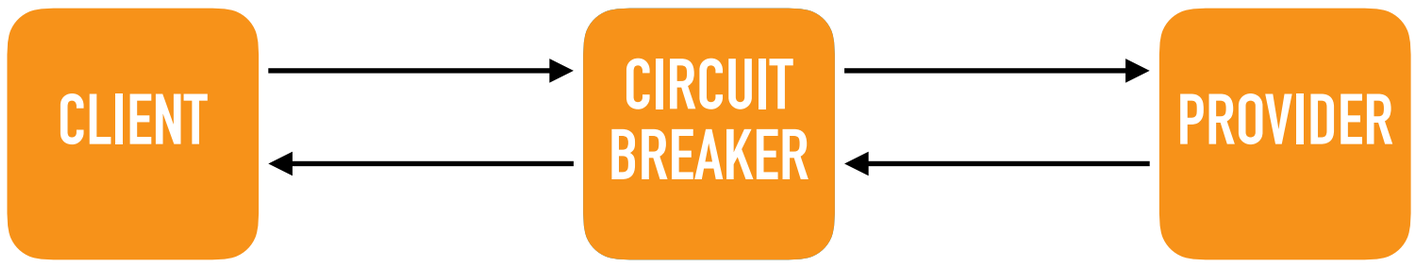
TRIAL

CIRCUIT BREAKER



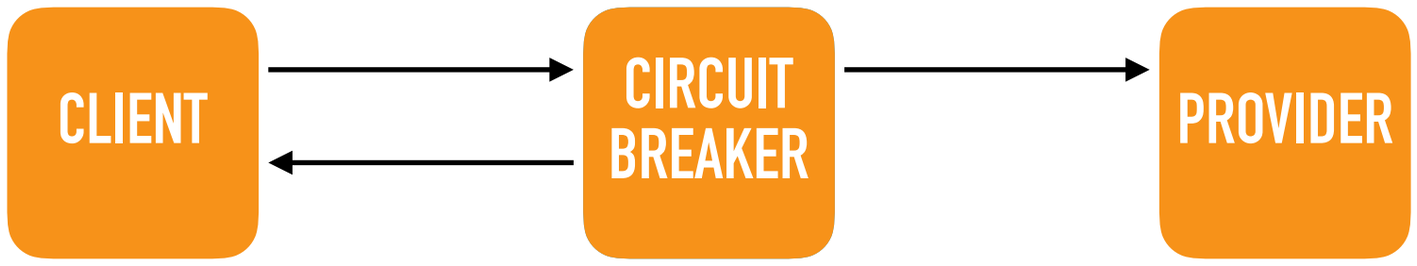
RECOVERY

CIRCUIT BREAKER



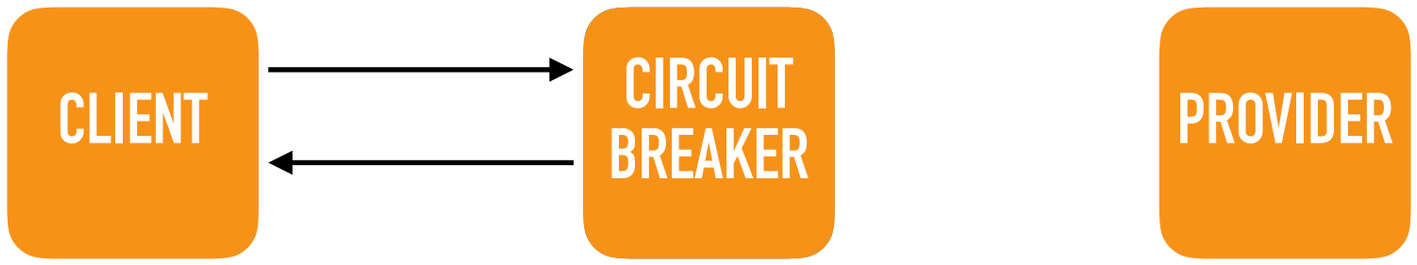
NORMAL

CIRCUIT BREAKER



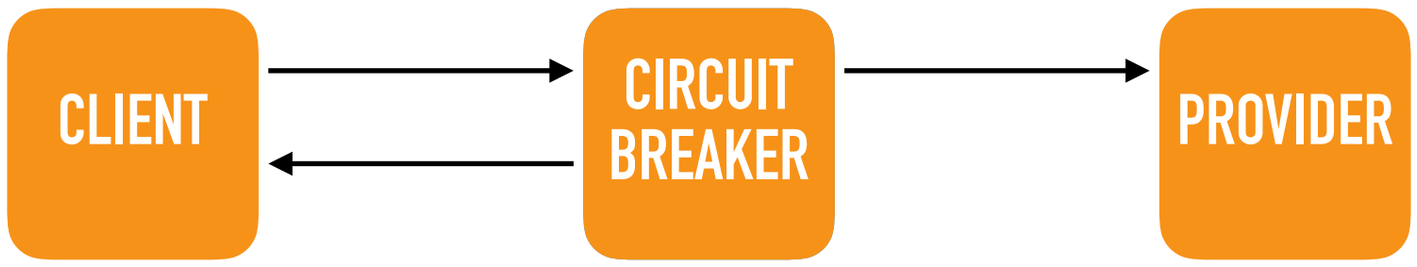
TIMEOUT

CIRCUIT BREAKER



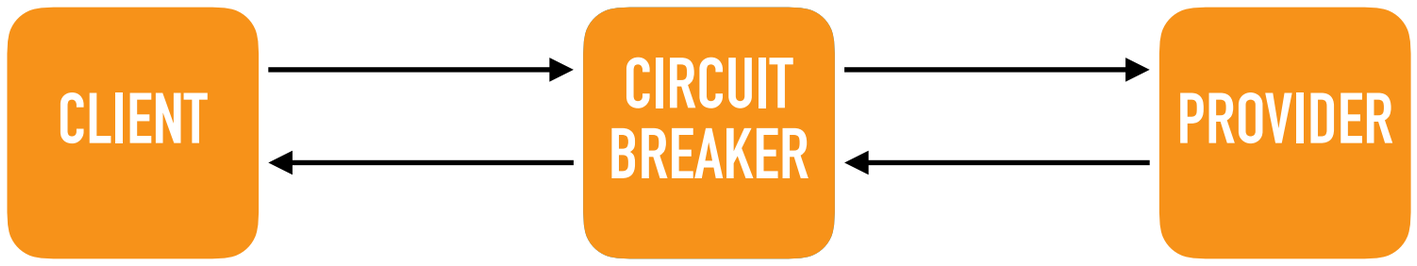
CIRCUIT OPEN

CIRCUIT BREAKER



TRIAL

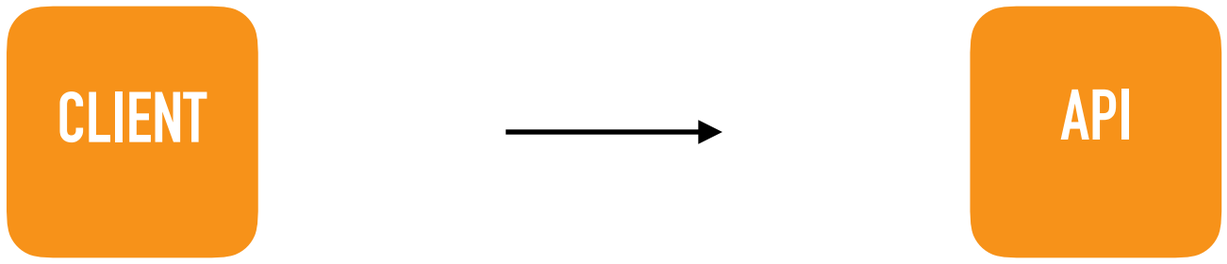
CIRCUIT BREAKER



RECOVERY

ASYNCHRONOUS

ASYNCHRONOUS



Source: Asynchronous patterns for Cloud Functions by Preston Holmes https://cloud.google.com/community/tutorials/cloud-functions-async

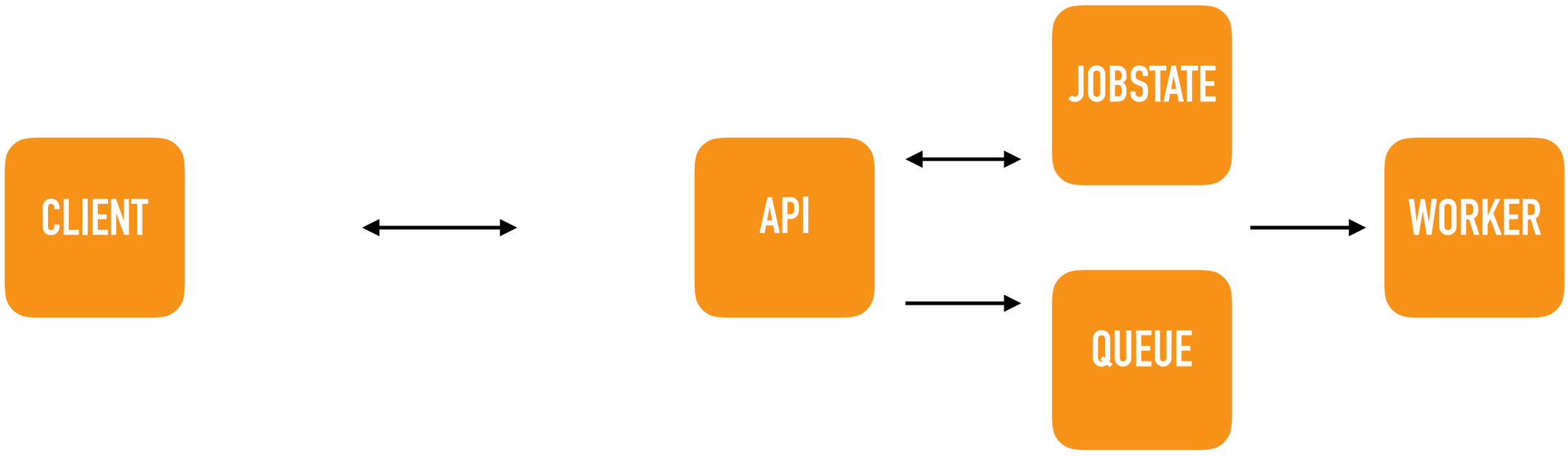
ASYNCHRONOUS

Source: Asynchronous patterns for Cloud Functions by Preston Holmes https://cloud.google.com/community/tutorials/cloud-functions-async

ASYNCHRONOUS

Source: Asynchronous patterns for Cloud Functions by Preston Holmes https://cloud.google.com/community/tutorials/cloud-functions-async

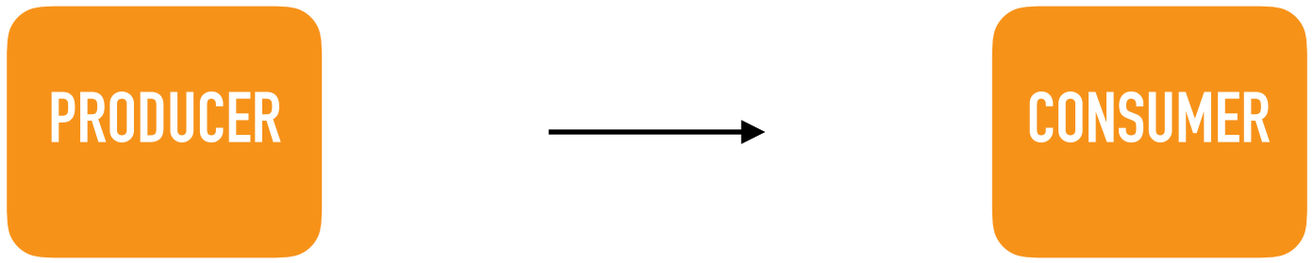
ASYNCHRONOUS



Source: Asynchronous patterns for Cloud Functions by Preston Holmes https://cloud.google.com/community/tutorials/cloud-functions-async

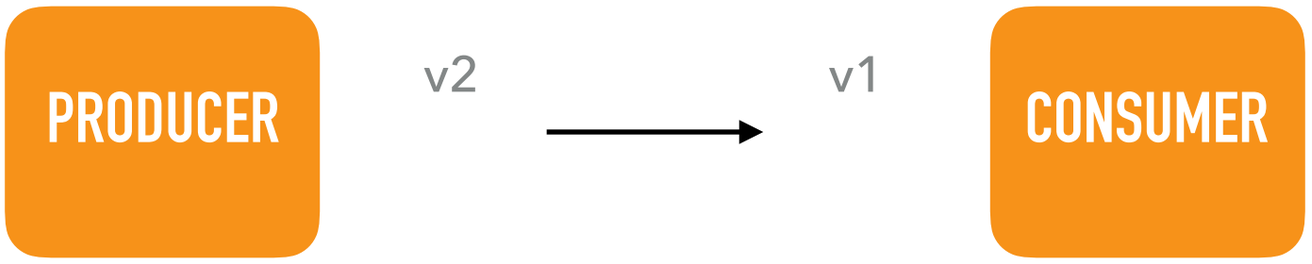
EVENT DRIVEN

EVENT DRIVEN



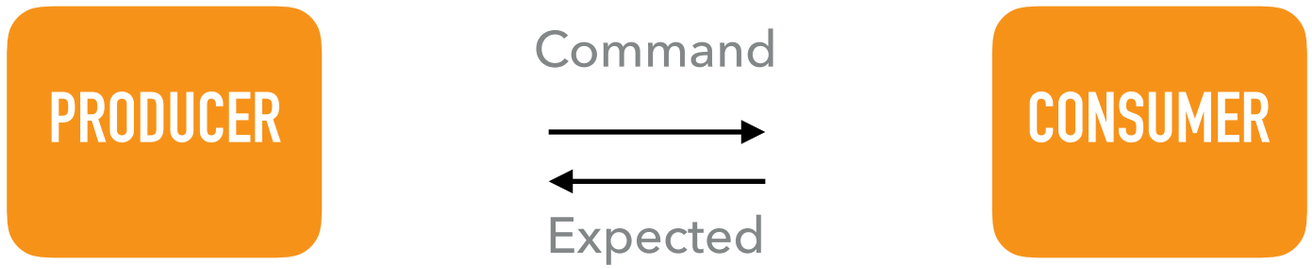
Source: Event Driven by Martin Fowler https://martinfowler.com/articles/201701-event-driven.html

EVENT DRIVEN



Source: Event Driven by Martin Fowler https://martinfowler.com/articles/201701-event-driven.html

EVENT DRIVEN



Source: Event Driven by Martin Fowler https://martinfowler.com/articles/201701-event-driven.html

QUALITY VS INNOVATION

SITE RELIABILITY ENGINEERING

- How much quality have we agreed upon?
- How much quality do we provide?
- How much quality do we want?

Source: Site Reliability Engineering ISBN: https://landing.google.com/sre/sre-book/toc/

QUANTIFY QUALITY

ERROR BUDGET

TOIL

TOIL

- Designate Engineer
  - Focus on incidents
  - Shields the team
  - Engineers solutions
  - Close collaboration with Product Owner


OBSERVABILITY


THE DAILY
FAKE NEWS
ECONOMY OF THE EUROPEAN UNION
WORLD BANK'S STOCK AT ALL-TIME HIGH
THE WORLD IS MESSED UP
FALSE POSITIVES

USER NOTIFICATIONS

ATTRIBUTION

- Sources are on bottom of the slides
- All pictures are from unsplash.com and their creators

FURTHER READING

Source: AWS Well-Architected Framework Whitepaper https://aws.amazon.com/architecture/well-architected/

STEFFAN NORBERHUIS

- Freelance Cloud & DevOps Consultant
- Twitter: @SNorberhuis
- steffan@norberhuis.nl

ANY QUESTIONS?