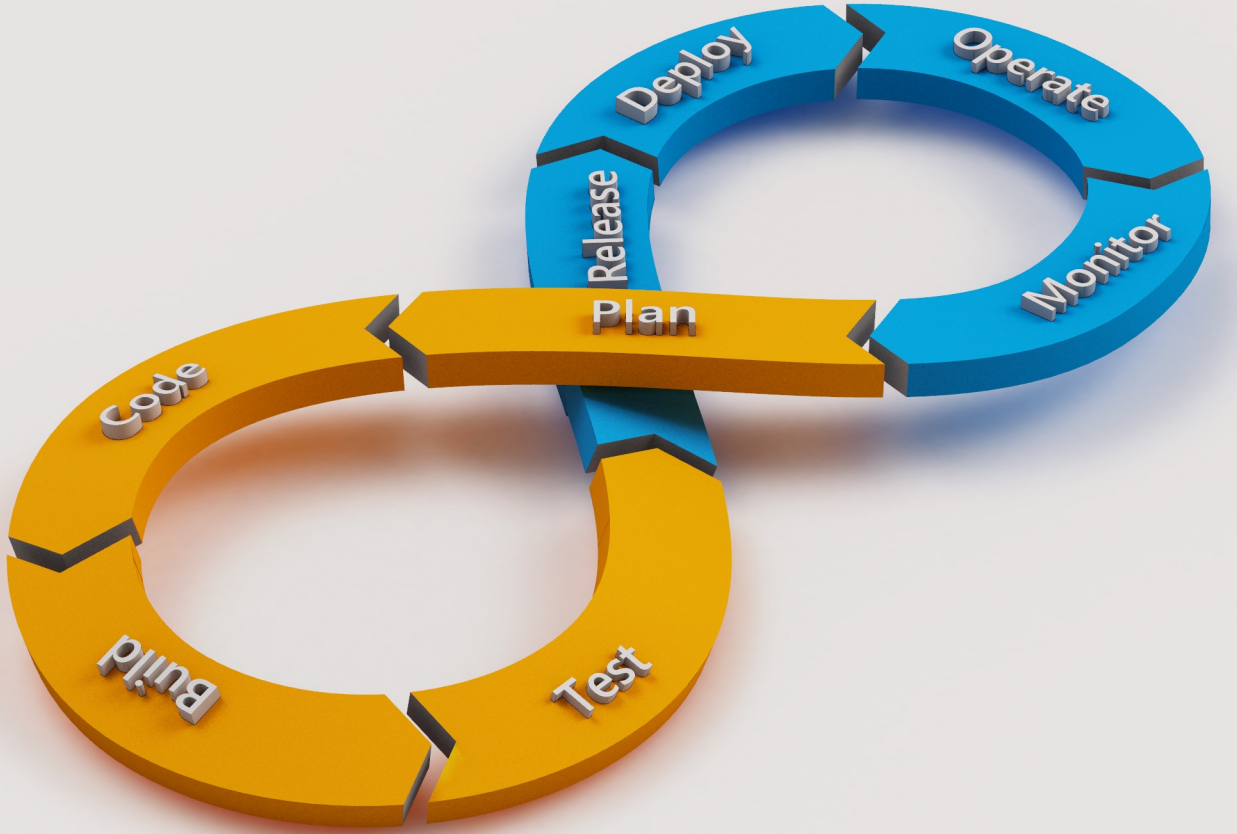
CI/CD In the Serverless Era

Efi Merdler-Kravitz

Serverless Hero, VP of Engineering, Lumigo

efi@lumigo.io

@TServerless



lumigo

- 100 % Serverless in production
- 100 % Serverless in CI/CD
- Tips to speed the flow and make it more robust
- Unique flows for phased deployment

lumigo

Some stats

~ 20K individual test runs per month

~ 30 deploys to production per week

~ 1H average deploy time to integration environment

~ 40m average time to run the tests

Who am I

AWS Serverless Hero
Head of Lumigo R&D
3rd company I'm using Serverless

Working at Lumigo in the past three years:

- SaaS platform for AWS serverless observability
- GA since Q1'2019

lumigo

Agenda

- The DevOps Infinity Loop - Revisited
- Our best practices

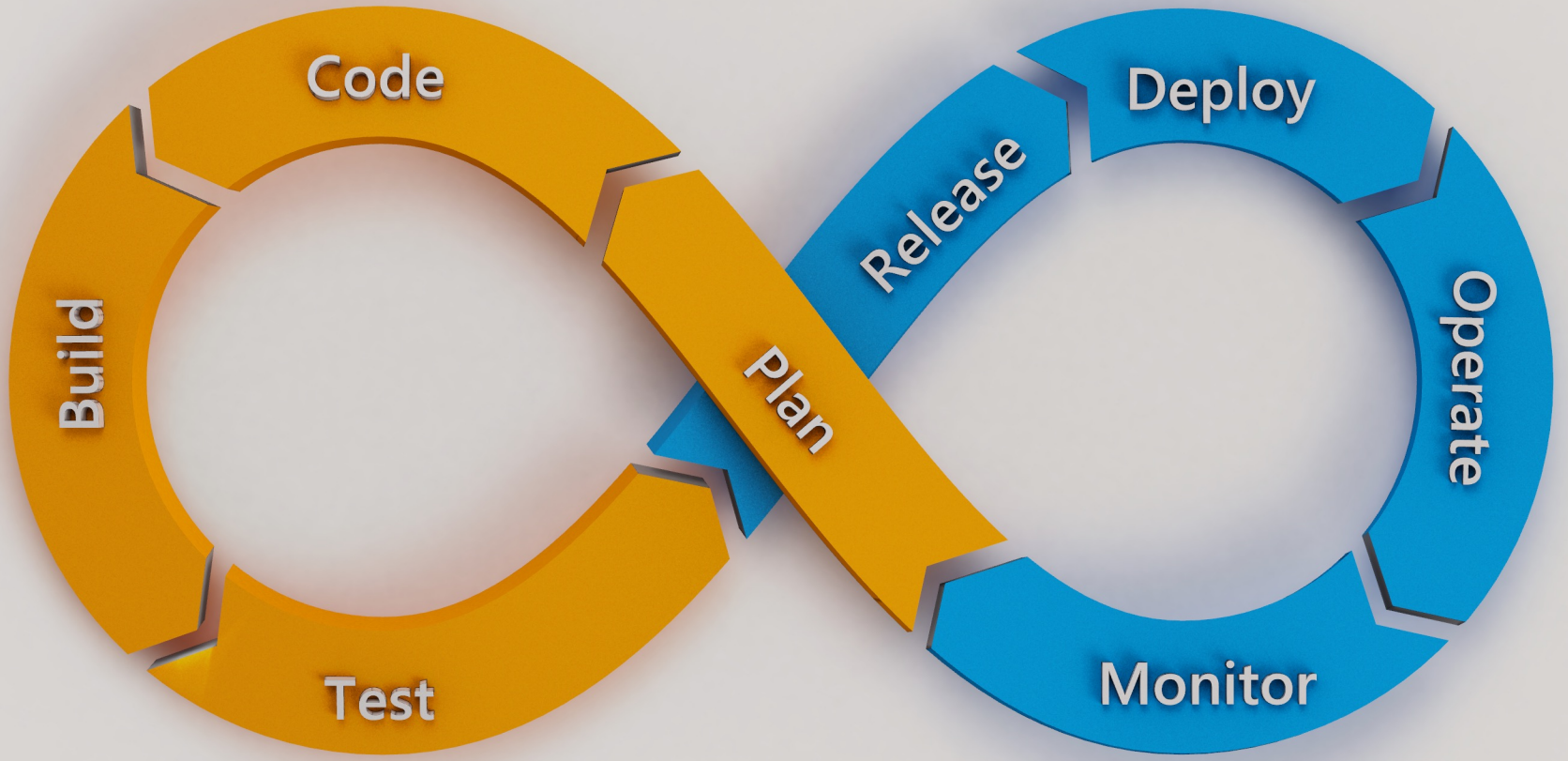
lumigo

Serverless is Different

DISTRIBUTED MICROSERVICES

- Hundreds of components
  - Can’t run locally
- Frequent deployments

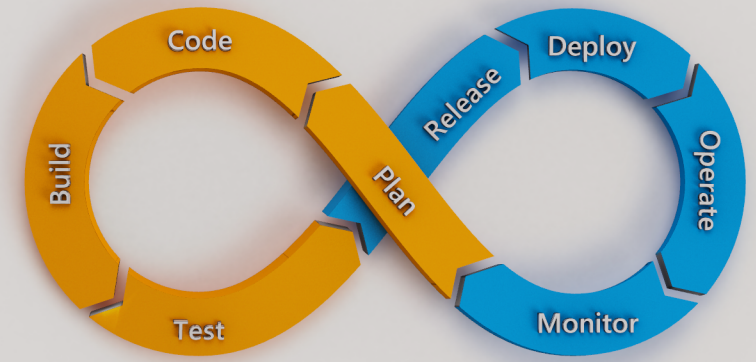
The Infinity Loop



lumigo

Guidelines

- Own environment
- Serverless services
- No dedicated QA or Ops



lumigo

Environments

Personal Environment

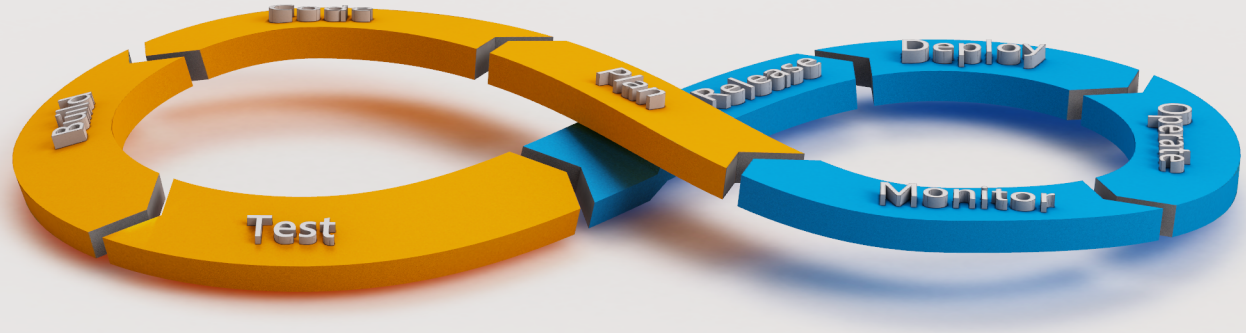
Integration Environment

Production

Local Development

Monitoring Environment

Our Stack



3.7 Released

Jira Software

cypress.io

GitHub

serverless

circleci

https://tinyurl.com/proactive-lumigo

MOCHA

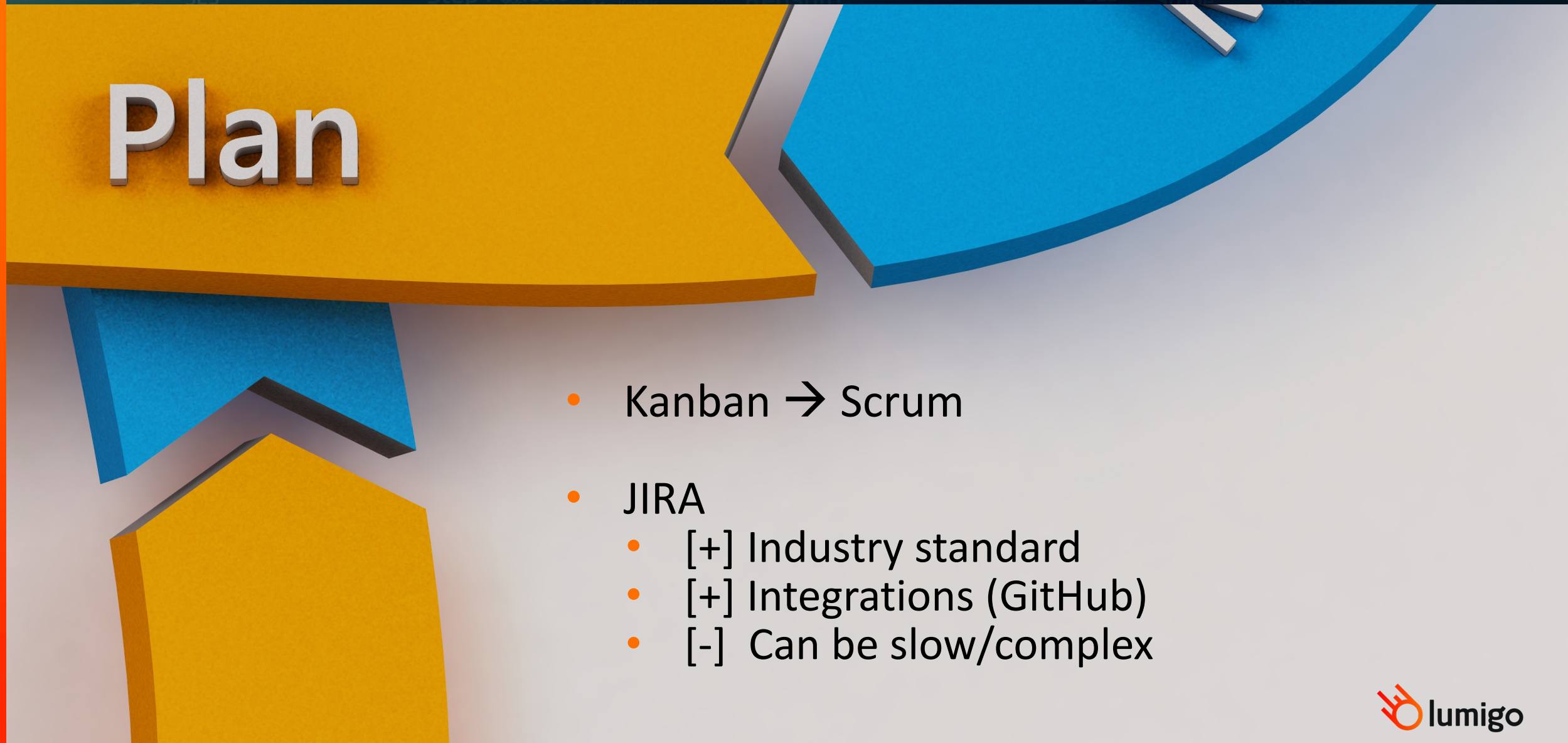
Codecov

pytest

node

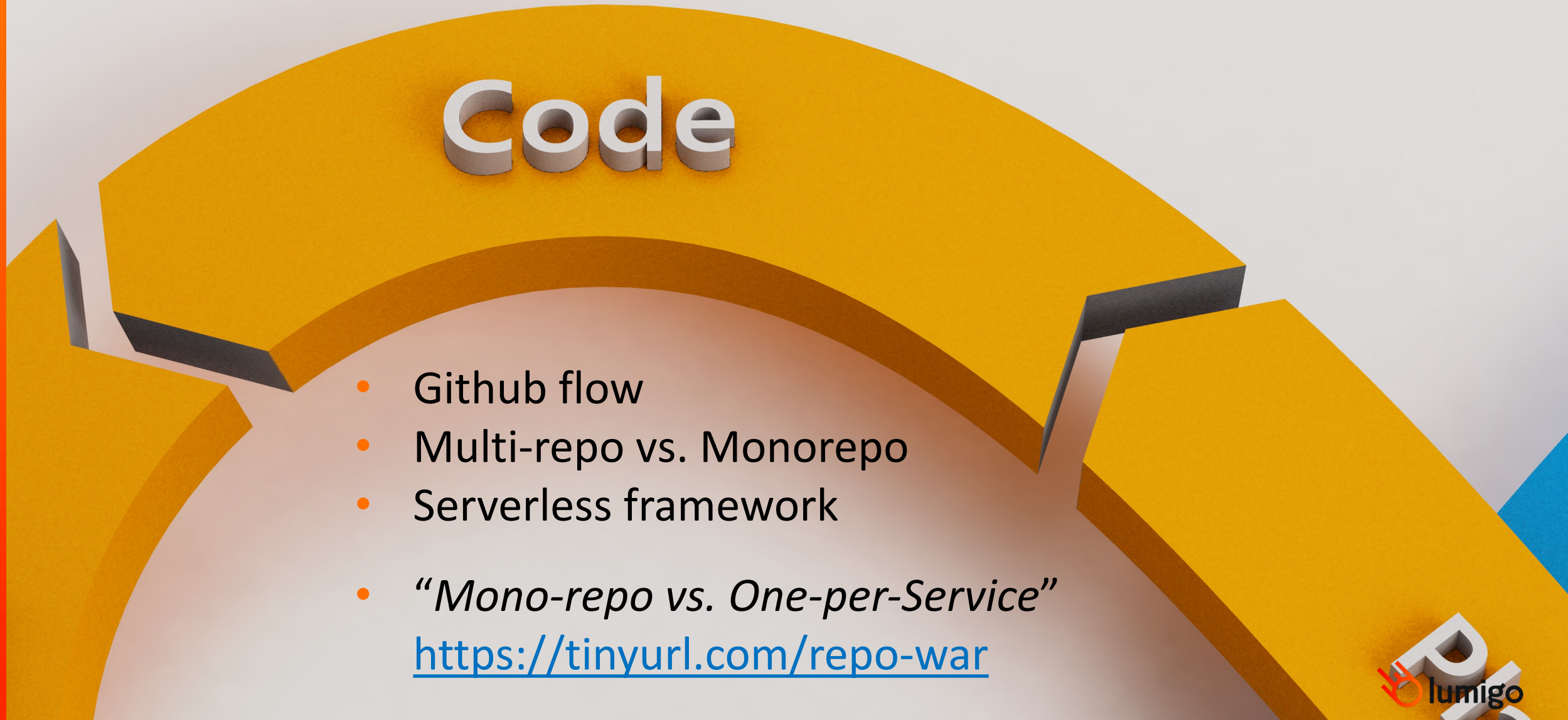
lumigo

Planning with JIRA



- Kanban → Scrum
- JIRA
  - [+] Industry standard
  - [+] Integrations (GitHub)
  - [-] Can be slow/complex

Coding with GitHub



- Github flow
- Multi-repo vs. Monorepo
- Serverless framework

- *"Mono-repo vs. One-per-Service"*
  https://tinyurl.com/repo-war

Coding

- YML per service
- Uber deploy
  gets the needed code

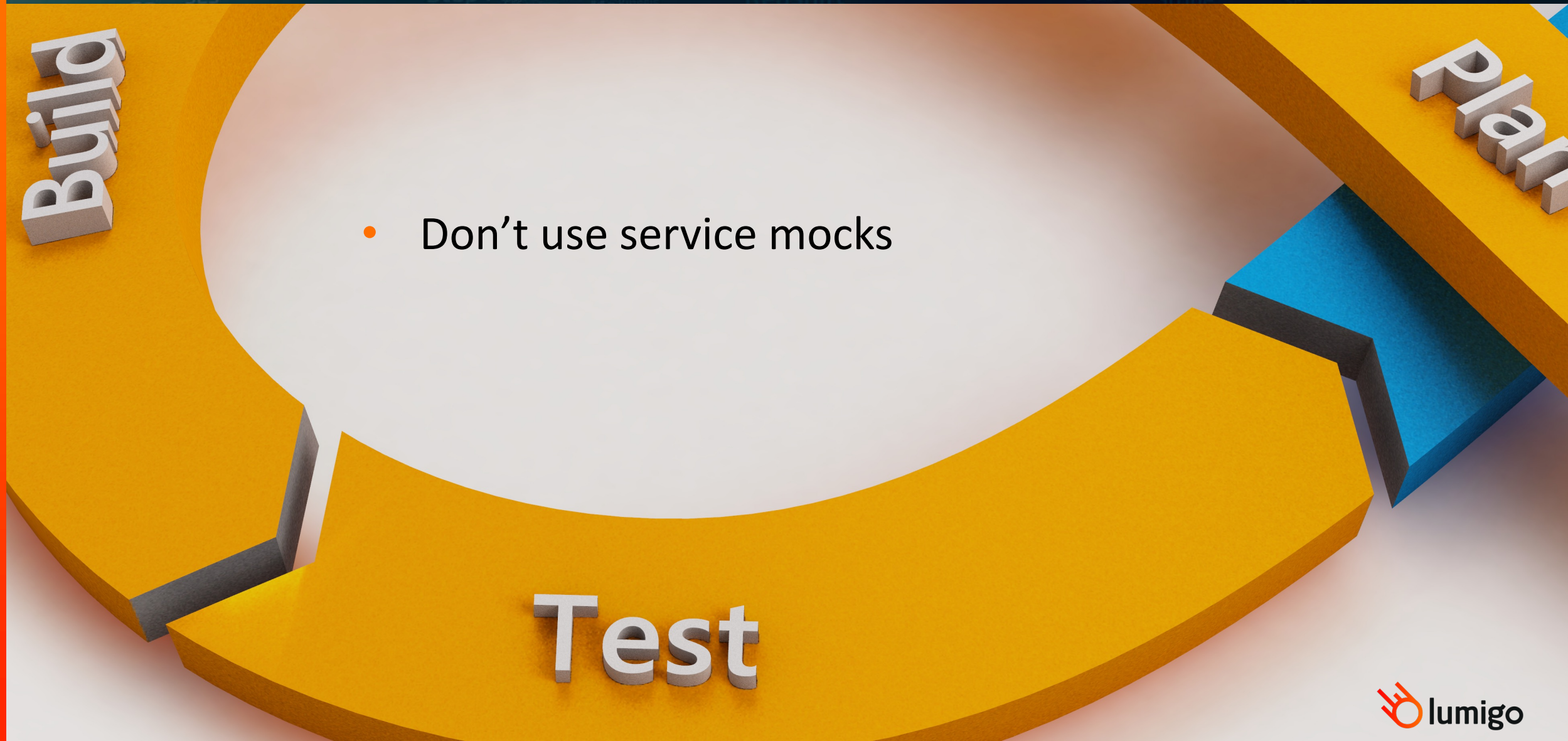
Test locally

- Unit tests
- Git pre-commit hooks

- *"Testing serverless from the trenches"*
  https://tinyurl.com/testing-serverless

lumigo

Unit tests vs Integration tests



- Don’t use service mocks

lumigo

Testing Stack

- Linting   Flake8 ESLint
- Static type analysis  : my[py]
- Unit testing   

lumigo

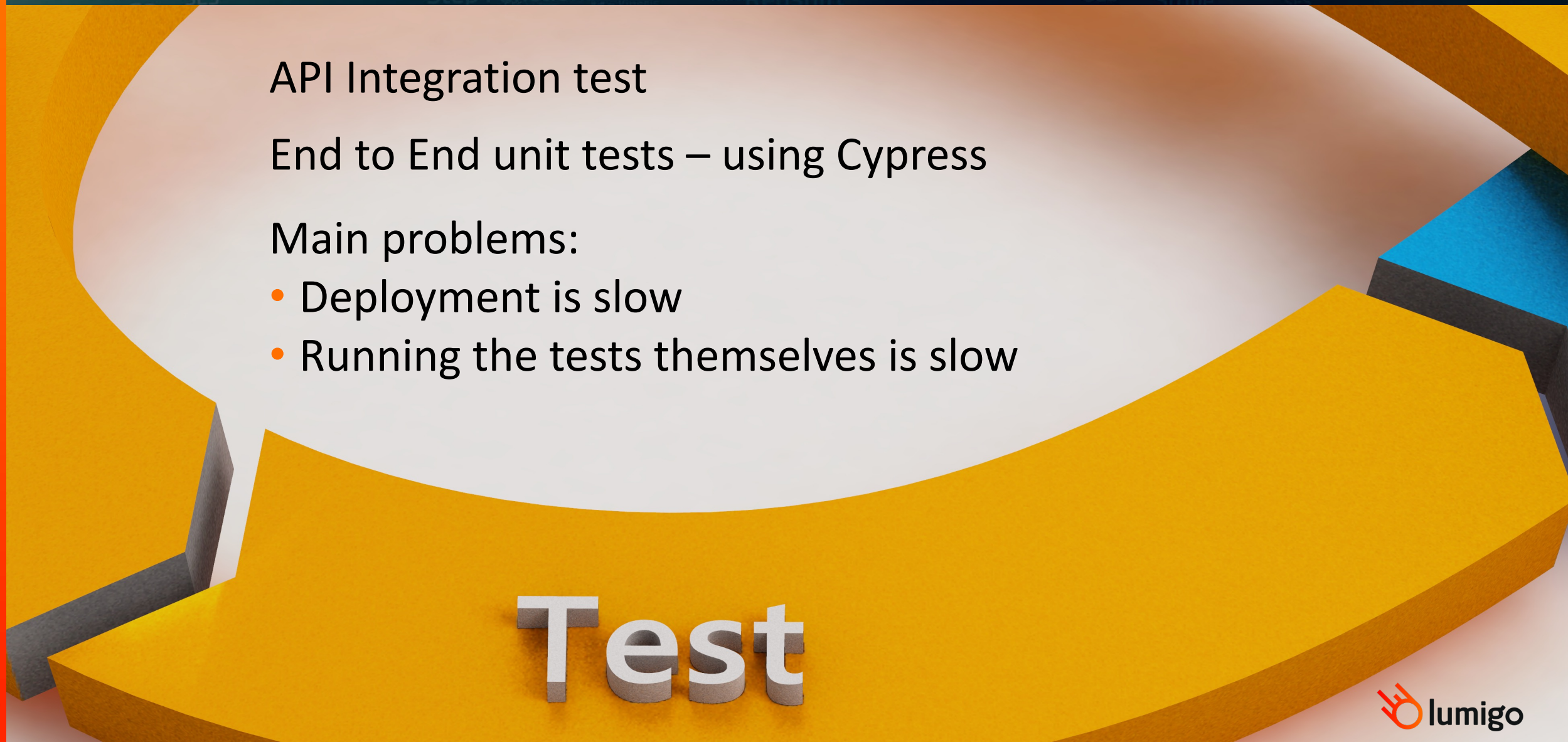
Integration testing

API Integration test

End to End unit tests – using Cypress

Main problems:

- Deployment is slow
- Running the tests themselves is slow



Integration testing

- Each test in a dedicate AWS account – Isolation
- Parallel running of all tests (hundreds) – Speed
- No additional cost – creating the environment cost nothing, just the execution – Serverless power

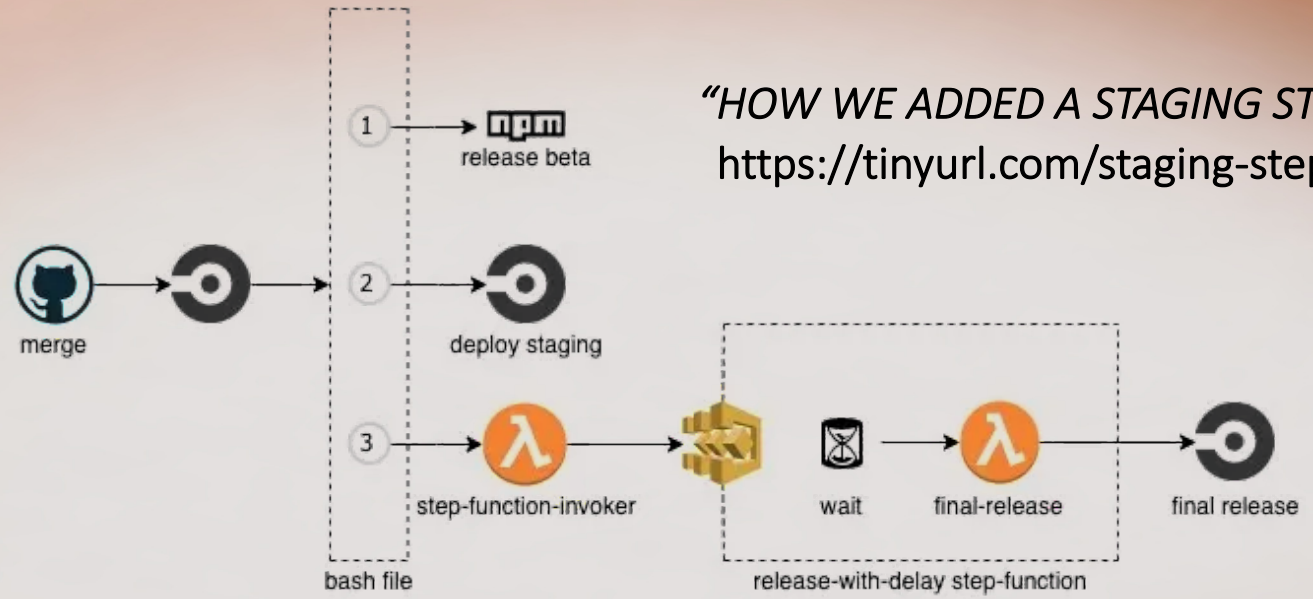
"SERVERLESS CONTINUOUS INTEGRATION IN THE ERA OF PARALLELISM"
https://tinyurl.com/it-parallel

lumigo

Integration testing

- Using CF tags to avoid redeployment
- Still no good solution for initial deployments

Testing critical components

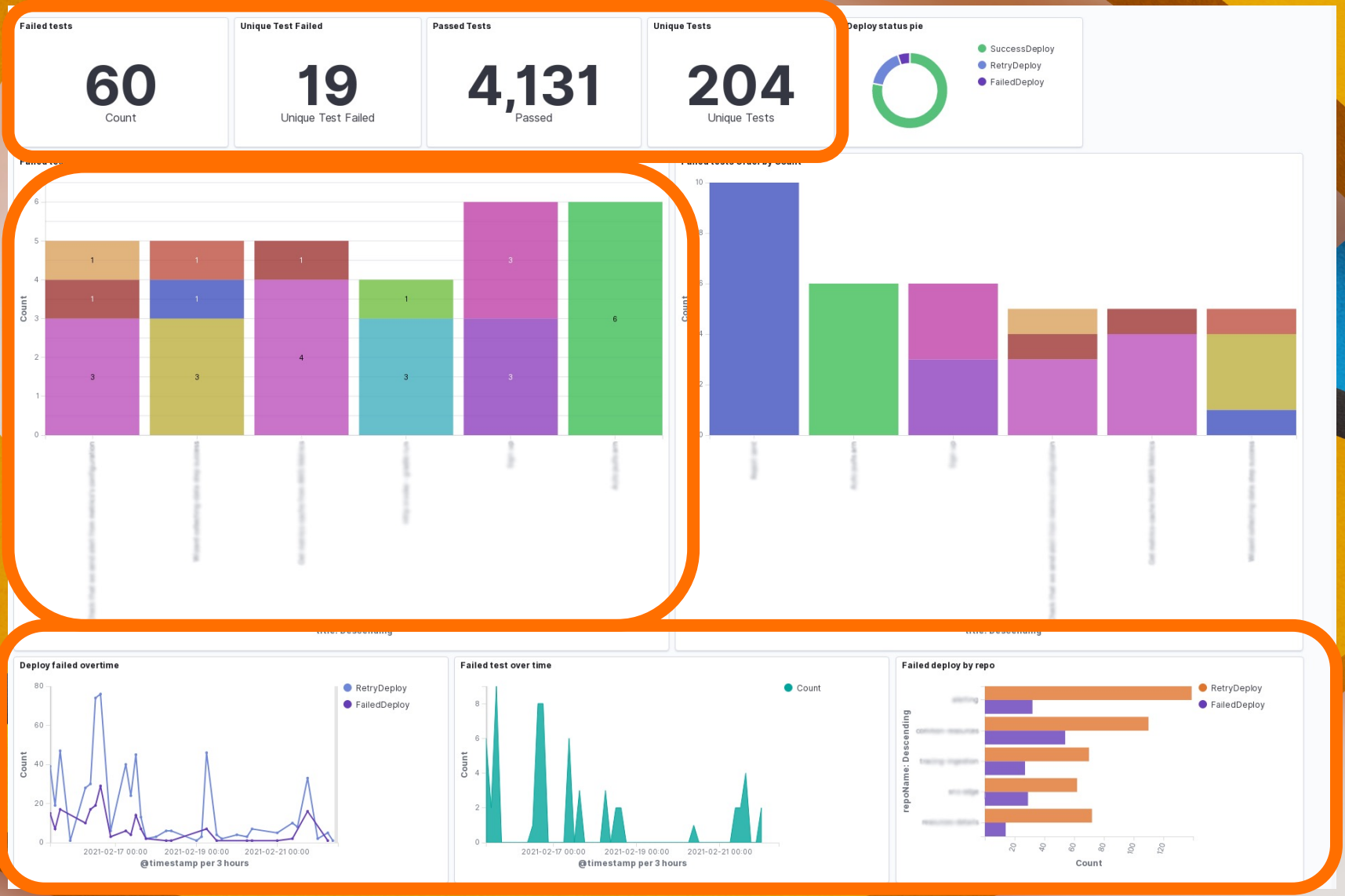


"HOW WE ADDED A STAGING STEP"
https://tinyurl.com/staging-step

Test

lumigo

Integration testing metrics



lumigo

Cleaning

- Really difficult!
- We use a combination of lumigo-cli and AWS-nuke

“Reusing internal AWS accounts: Challenges and lessons learned”
https://tinyurl.com/clean-aws

lumigo

Release Gates

- Code Review
- Code Coverage
- Linting + Type testing
- Tests
- Merge to master
  → deploy to prod

Release Gate

Changes approved
1 approving review by reviewers with write access. Learn more.
Hide all reviewers

1 approval

uri-p approved these changes

All checks have passed
5 successful checks
Hide all checks

ci/circleci: e2e-test — Your tests passed on CircleCI! Details

ci/circleci: integration-test — Your tests passed on CircleCI! Required Details

ci/circleci: test — Your tests passed on CircleCI! Required Details

codecov/patch — 98.36% of diff hit (target 97.39%) Details

codecov/project — 97.58% (+0.18%) compared to e8bcfd2 Details

This branch has conflicts that must be resolved
Use the web editor or the command line to resolve conflicts.
Resolve conflicts

Conflicting files

`requirements.txt`

Monitoring Serverless

lumigo

Serverless: “Monitoring in the Dark”

Use our own dog food

lumigo

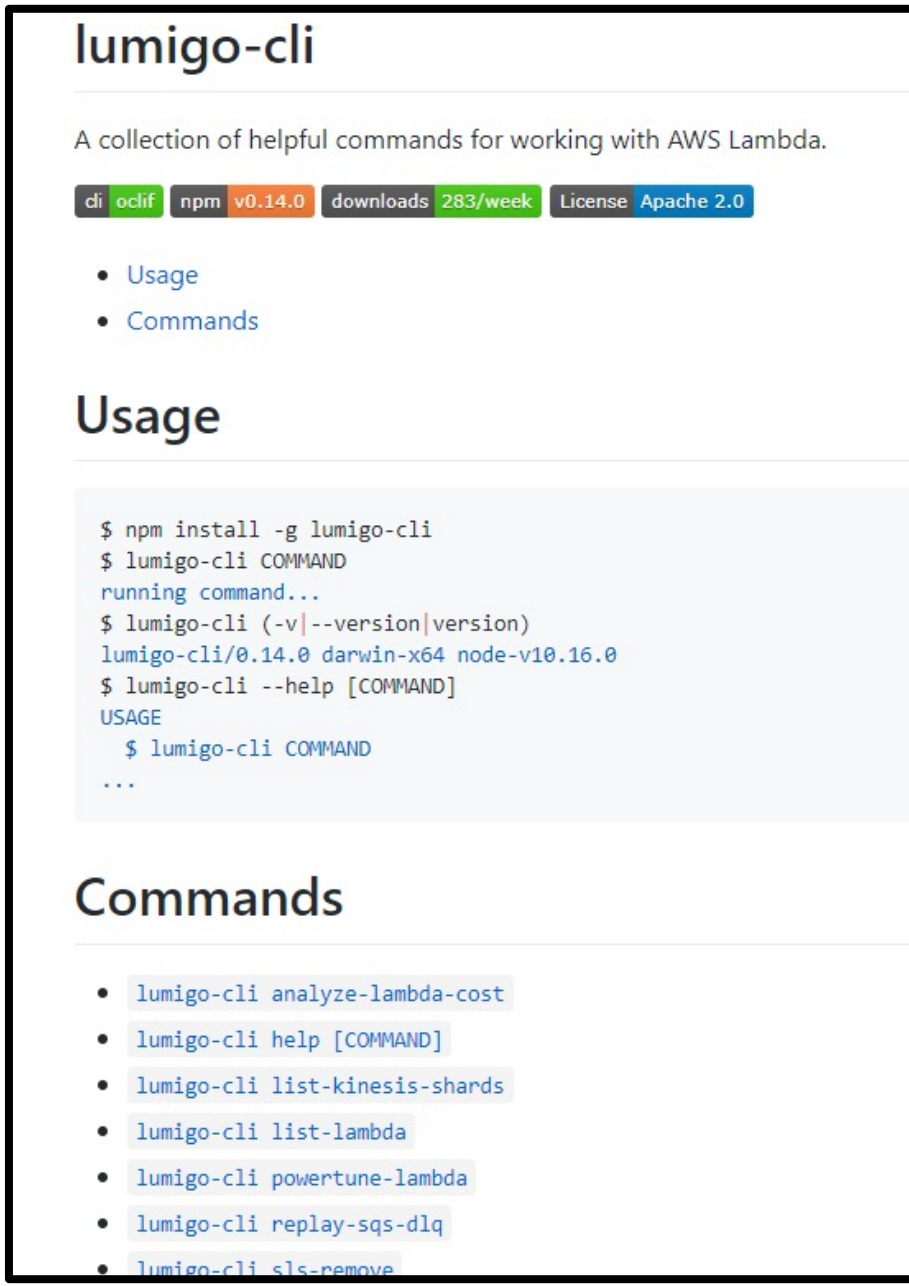
Takeaways

- Use the power of Serverless (Parallel testing)
- Integration tests are the king
- Automate your deployment process so it’s easy for developers

Most used CLI commands by our users:

- Switch account
- Clear account
- Analyze cold starts

https://github.com/lumigo-io/lumigo-CLI



lumigo-cli

A collection of helpful commands for working with AWS Lambda.

cli oclif | npm v0.14.0 | downloads 283/week | License Apache 2.0

- Usage
- Commands

Usage

```
$ npm install -g lumigo-cli
$ lumigo-cli COMMAND
running command...
$ lumigo-cli (-v|--version|version)
lumigo-cli/0.14.0 darwin-x64 node-v10.16.0
$ lumigo-cli --help [COMMAND]
USAGE
  $ lumigo-cli COMMAND
...
```

Commands

- `lumigo-cli analyze-lambda-cost`
- `lumigo-cli help [COMMAND]`
- `lumigo-cli list-kinesis-shards`
- `lumigo-cli list-lambda`
- `lumigo-cli powertune-lambda`
- `lumigo-cli replay-sqs-dlq`
- `lumigo-cli sls-remove`

THANK YOU !

Questions?

Efi Merdler-Kravitz

VP of Engineering

efi@lumigo.io

@TServerless

lumigo