ScrumAlliance®
Transforming the World of Work

kelsus INC.
LEAN CUSTOM SOFTWARE

Agility on Steroids with a Docker-based CI/CD pipeline on AWS

Jon Christensen | July 19, 2018

We were using Heroku or Elastic Beanstalk

Users were the first to notice issues

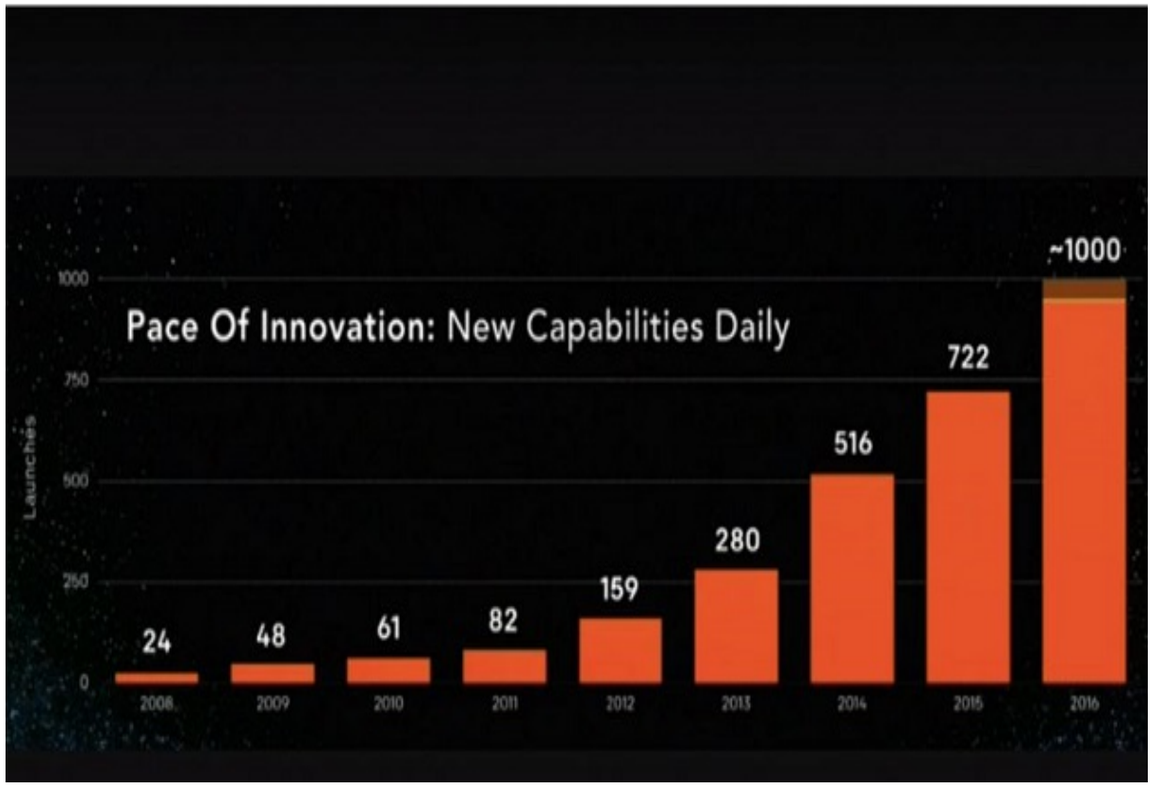
We had minimal test coverage

We had disorganized logs

We were hackers

This wasn’t going to cut it at scale

- There are too many choices.
- AWS alone adds 5 new features a day.
- So the best way to start CI/CD is gradually.
- Just like your software itself, CI/CD is always a work in progress.



ScrumAlliance®

I truly believe that one of the best benefits of Docker is that it forces less experienced developers on teams to confront head-on some of the things they don't know about how computers work and learn them and incorporate them into their body of knowledge

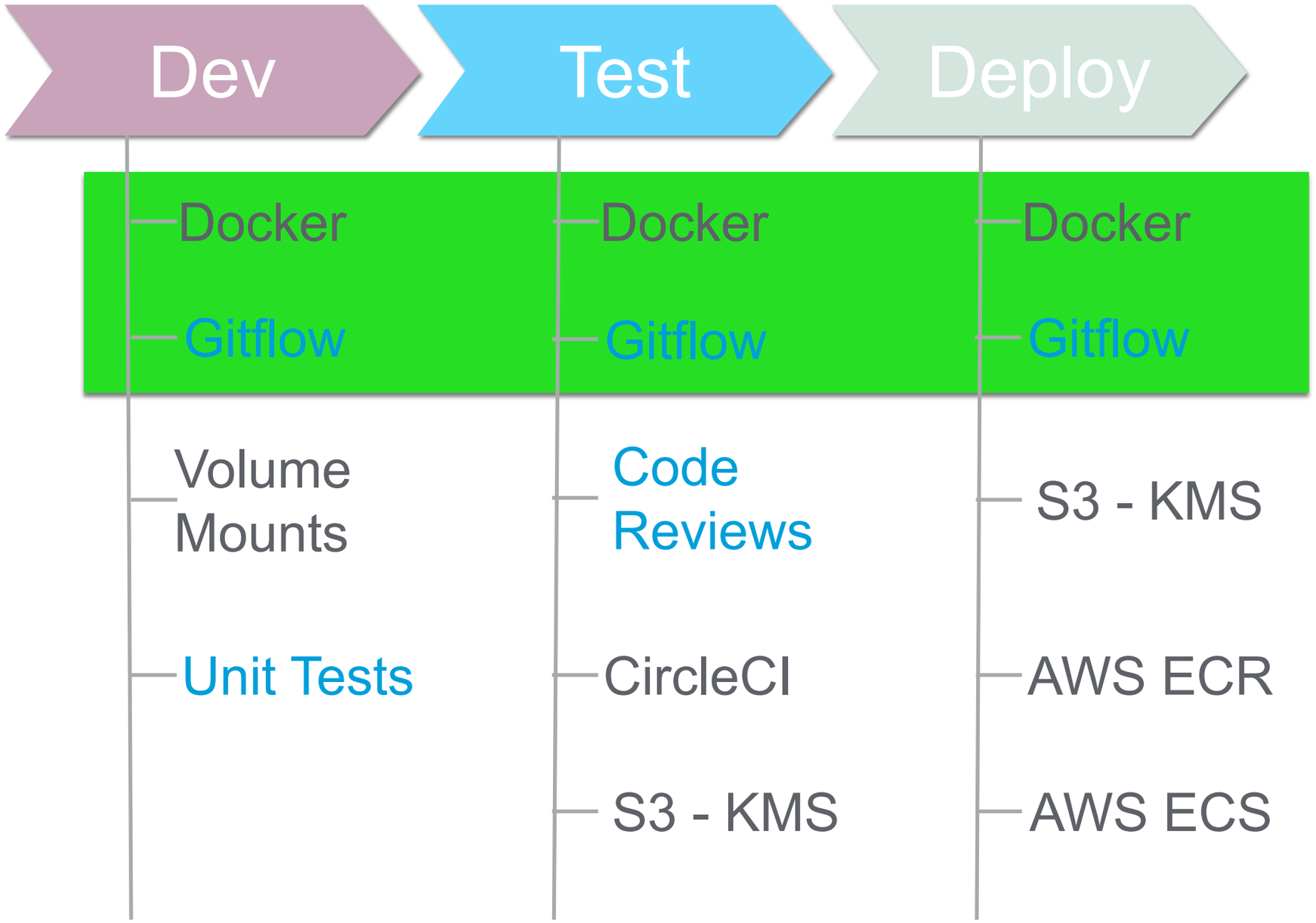
"True" continuous deployment requires no manual intervention or human involvement.

Requires two pillars:

1. Comprehensive, automated integration tests with broad coverage
2. The ability to rollback quickly (ideally automated) in case a bad build gets deployed (because your verification wasn't complete enough)

ScrumAlliance®

Current CI/CD Tools and Processes



Volume Mounts for Hot Reloads

```
version: "2"

networks:
  docker-dev:
    external:
      name: dev

services:
    api:
        build: api/
        env_file: .env
        networks:
          default: {}
          docker-dev:
            aliases:
              - my-restful-service
        volumes:
            - ./shared:/var/www/my-restful-service/shared
        ports:
            - "8080:80"
        links:
            - db

    db:
        build: db/
        env_file: .env
        ports:
            - "32099:5432"
        volumes_from:
            - data

    data:
        image: postgres
        restart: "no"
        command: "true"
        volumes:
            - /var/lib/postgresql

    test:
        build: api/
        environment:
            APP_ENV: test
        command: "./run_tests_docker.sh"
        volumes:
            - ./shared:/var/www/my-restful-service/shared
        ports:
            - "8080:80"
        links:
            - test-db

    test-db:
        build: db/
        ports:
            - 5432
```

```
  ##########################
  # The below container is provided
  # as a developer convenience.
  # It mounts the host volume at the
  # project root (so that both container and
  # host are reading same files). This
  # allows for hot reloading of code changes.
  ##########################

  api-dev:
    build: api/
    env_file: .env
    volumes:
      - .:/var/www/my-restful-service
    ports:
      - "8080:80"
    links:
      - db
```

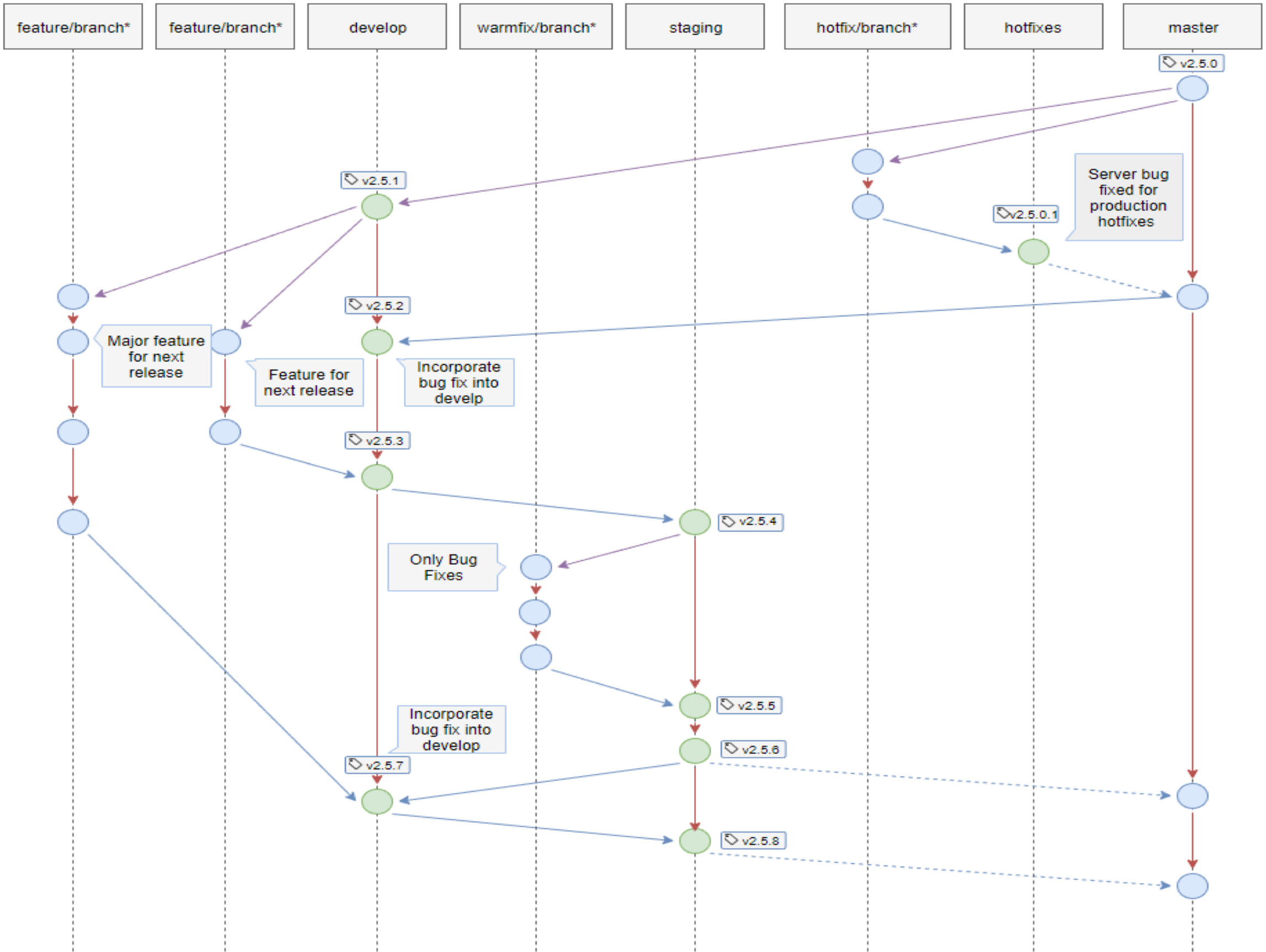
ScrumAlliance®

Testing has to be a moral mandate

In order to succeed at always writing tests, the organization as a whole must treat *not* testing as ***unthinkable***

ScrumAlliance®

Kelsus Git Flow



ScrumAlliance®

Formerly a walkthrough of a CircleCI file

- Primarily define a build
  - What Docker image to start from?
  - Set some environment variables (like how to get access to the code)
- Then define some tests
  - Run tests with docker compose
  - Your tests themselves and where they live will depend on things like your language and framework
  - Collect artifacts (coverage logs, pass/fail, actual images for later deployment)
- And lastly define a deployment if the tests pass
  - Because we're using gitflow we will have a script that will tell our services to use a new docker image if we've checked into the develop or staging branch

ScrumAlliance®

Take care of secrets

This used to be a slide showing how to do secrets management with AWS S3.

But the important thing is to not use git or environment variables for your code secrets.

AWS has a new secrets manager that we're likely to move to.

ScrumAlliance®

Why ECS?

excerpt from hangops slack

Thread
luckymike, charlieo, and 2 others

luckymike Today at 9:31 AM
in #aws

If it's anything like ECS, probably go right on doing what they're doing today.

8 replies

jonxtensen 45 minutes ago
There an underlying assumption here about ECS being bad. Would you mind telling me what you don't like about it? So far we've had good success with it but it might just be that we don't have experience at the kinds of scale or complexity you're looking at.

luckymike 5 minutes ago
It's not really a bad. More that AWS's software (e.g. Elasticache, ES, ECS) implementations tend to be fairly narrow and lose a lot of power in exchange for dropping some operational complexity.

jonxtensen 4 minutes ago
Right on. I'd definitely agree with that

luckymike 3 minutes ago
Its admittedly been a few years since I used it. It felt like it was some Docker glommed onto EC2 instances.

luckymike 2 minutes ago
That in of itself isn't bad, but ECS paled in comparison to the autoscaling group it ran on top of.

charlieo 1 minute ago
It was horrible up until about a year and a half ago. It's actually pretty good now. The big thing about ECS is that it's actually incredibly flexible. ALB integration for rolling deploys is the biggest benefit IMO. Also it's free outside of EC2 / ALB costs.

myoung34 < 1 minute ago
ecs is improving, id def re-evaluate if you havent used it in ~12mo

myoung34 < 1 minute ago
the ecs team is very very responsive to community suggestions and is evolving it from docker-compose-in-aws to something bigger

Reply...

jonxtensen 45 minutes ago
There an underlying assumption here about ECS being bad. Would you mind telling me what you don't like about it? So far

charlieo 1 minute ago
It was horrible up until about a year and a half ago. It's actually pretty good now. The big thing about ECS is that it's actually incredibly flexible. ALB integration for rolling deploys is the biggest benefit IMO. Also it's free outside of EC2 / ALB costs.

myoung34 < 1 minute ago
ecs is improving, id def re-evaluate if you havent used it in ~12mo

ScrumAlliance®

CI/CD is always a work in progress

You're always about half done

We still need to work on

- Improving database migrations
- Getting off of bash scripts
- Updating how we store secrets
- Look at AWS Code Pipeline
- Better tests — always better tests

ScrumAlliance®

A final agile principal to work into the talk

Agile principle number 5: Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

Our motivated developers do seem to love being able to dig deep in AWS and Docker

ScrumAlliance®

Thank you!

— Jon Christensen

Kelsus Inc

www.kelsus.com
www.prodockertraining.com
@jonxtensen