Next Gen Deployments at Skillshare

Building extendable and composable pipelines using Codefresh

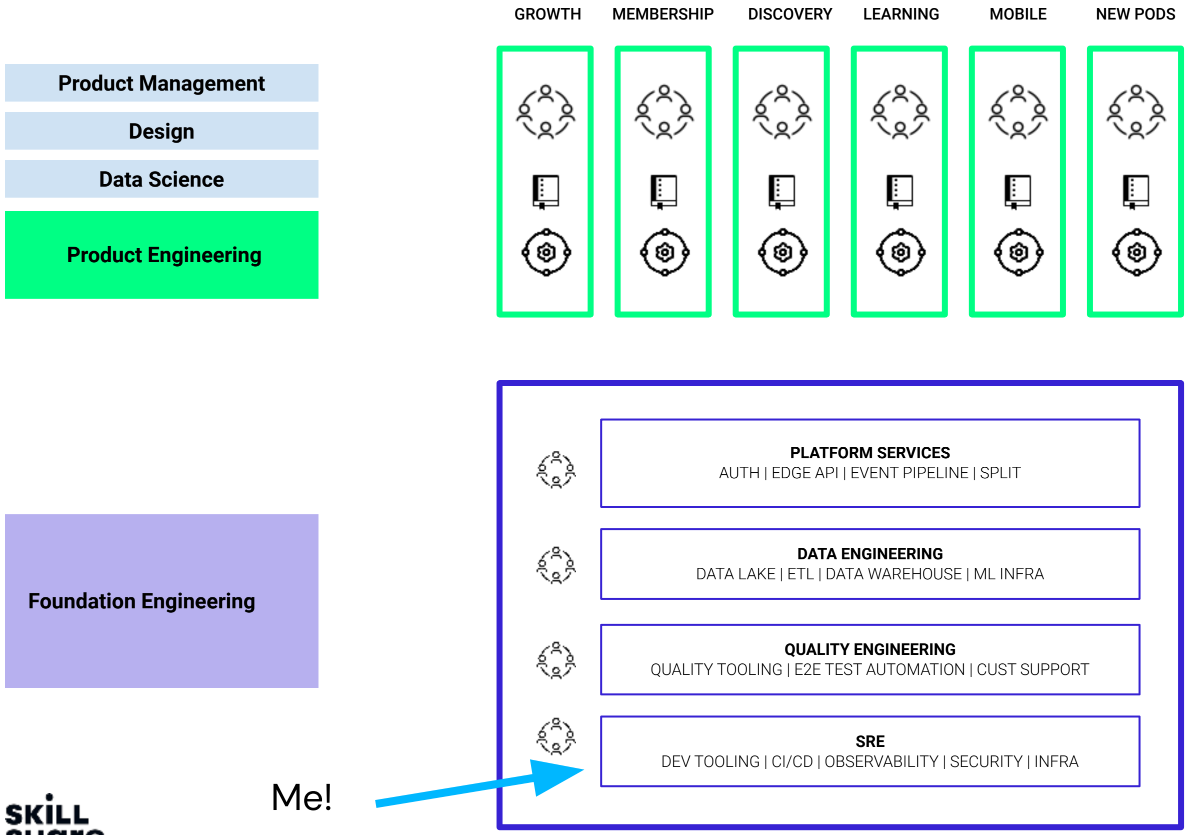
by Adam Hawkins - Staff SRE

SKILL SHare.

SKILL SHARE.

SKILL SHARE.

Internal Pod Structure



GROWTH | MEMBERSHIP | DISCOVERY | LEARNING | MOBILE | NEW PODS

Product Management

Design

Data Science

Product Engineering

Foundation Engineering

PLATFORM SERVICES
AUTH | EDGE API | EVENT PIPELINE | SPLIT

DATA ENGINEERING
DATA LAKE | ETL | DATA WAREHOUSE | ML INFRA

QUALITY ENGINEERING
QUALITY TOOLING | E2E TEST AUTOMATION | CUST SUPPORT

SRE
DEV TOOLING | CI/CD | OBSERVABILITY | SECURITY | INFRA

Me!

- ~35 engineers
- ~10 services
- NodeJS & PHP
- AWS, K8s, Helm

SKILL SHARE.

About Me

- ~10 years experience
- Currently Staff SRE
- Enable high velocity CD at Skillshare
- @adman65 on Twitter
- Smallbatches.fm podcast

Business Problem

The Business Problem

How can we support a growing engineering team?

1 **Secrets Outside SCM**

Developers had to go to GUIs to manage secrets then connect back to apps

2 **No Pipelines as Code**

All pipelines were managed through a GUI; no way to test changes on a branch

3 **No Multi-process Support**

Current tool did not gel well with 12 factor app model

4 **No Extensibility**

We can’t wait for the vendor to support us because we can move 10x faster

Where We Started

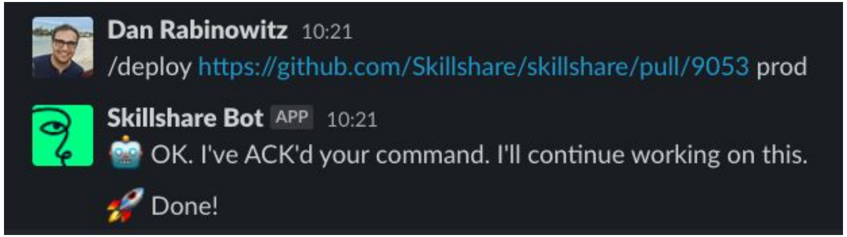
A short history lesson

T-Zero

- No unified deployment pipelines
- No pre-flight checks or smoke tests
- Every service is different
- No standard way to rollback prod

T-1: Unified CD

PIck a deploy tool; standardized k8s manifests; deploy through chatops



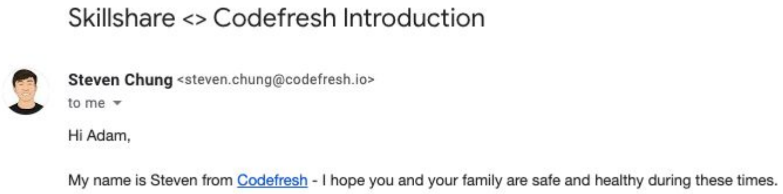
Dan Rabinowitz 10:21
/deploy https://github.com/Skillshare/skillshare/pull/9053 prod

Skillshare Bot APP 10:21
🤖 OK. I've ACK'd your command. I'll continue working on this.

🚀 Done!

T-2: One Year Later

We're stable but bumping up against those problems. It's clear we need a different tool, but what?



Skillshare <> Codefresh Introduction

Steven Chung <steven.chung@codefresh.io>
to me

Hi Adam,

My name is Steven from Codefresh - I hope you and your family are safe and healthy during these times.

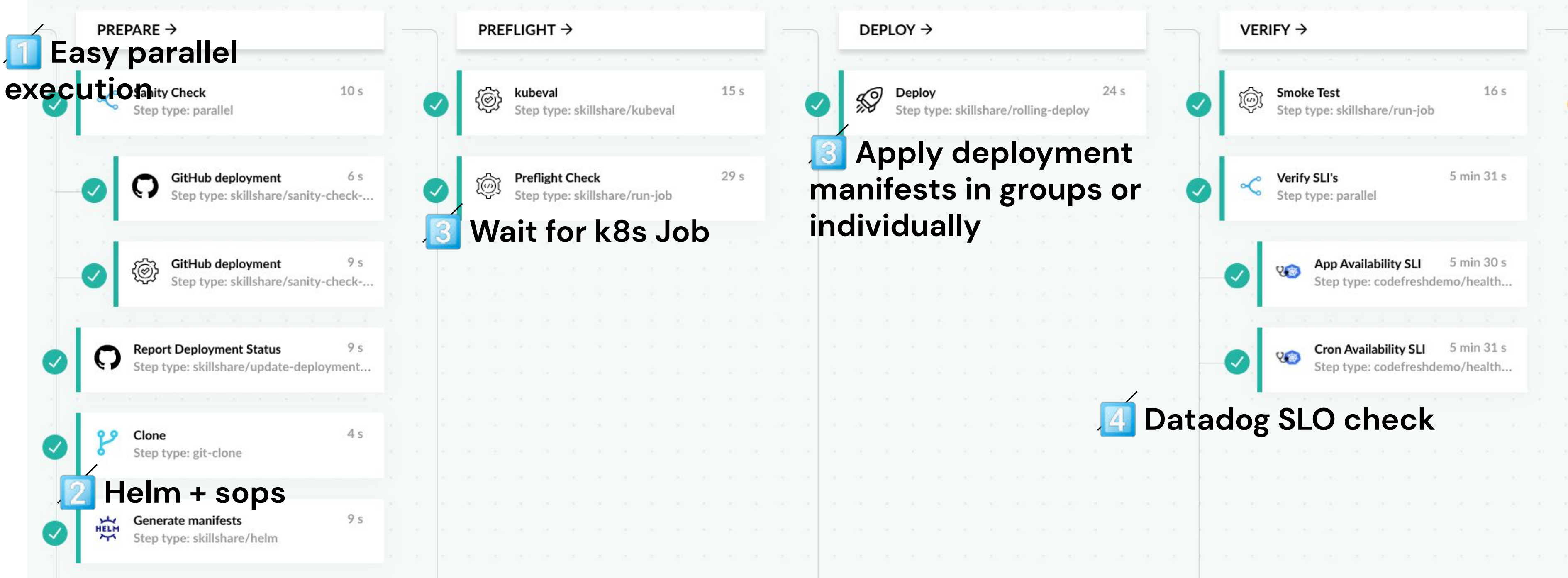
T-3: Meet Codefresh

Demo looks like it will solve all our problems, but we need proof.

🚀 Onward to a PoC

Proove support for our current and next requirements

The Proof-of-Concept Pipeline



PREPARE →

- Sanity Check — Step type: parallel — 10 s
  - GitHub deployment — Step type: skillshare/sanity-check-... — 6 s
  - GitHub deployment — Step type: skillshare/sanity-check-... — 9 s
- Report Deployment Status — Step type: skillshare/update-deployment... — 9 s
- Clone — Step type: git-clone — 4 s
- Generate manifests — Step type: skillshare/helm — 9 s

PREFLIGHT →

- kubeval — Step type: skillshare/kubeval — 15 s
- Preflight Check — Step type: skillshare/run-job — 29 s

DEPLOY →

- Deploy — Step type: skillshare/rolling-deploy — 24 s

VERIFY →

- Smoke Test — Step type: skillshare/run-job — 16 s
- Verify SLI's — Step type: parallel — 5 min 31 s
  - App Availability SLI — Step type: codefreshdemo/health... — 5 min 30 s
  - Cron Availability SLI — Step type: codefreshdemo/health... — 5 min 31 s

1️⃣ Easy parallel execution

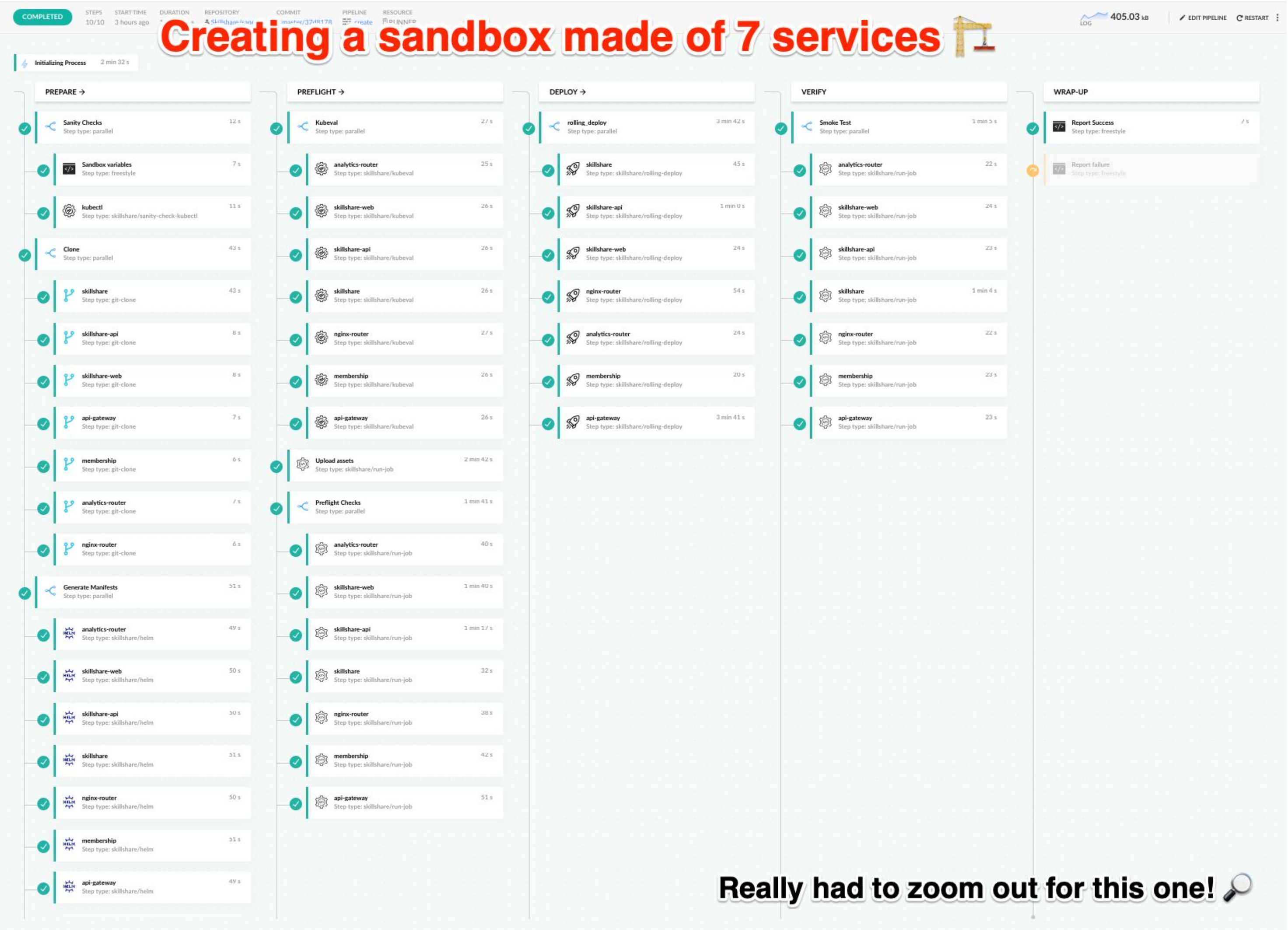
2️⃣ Helm + sops

3️⃣ Wait for k8s Job

3️⃣ Apply deployment manifests in groups or individually

4️⃣ Datadog SLO check

Creating a sandbox made of 7 services 🏗️



COMPLETED

STEPS 10/10 · START TIME 3 hours ago · DURATION · REPOSITORY Skillshare/sano · COMMIT master/37d8178 · PIPELINE create · RESOURCE RUNNER

LOG 405.03 kB · EDIT PIPELINE · RESTART

Initializing Process 2 min 32 s

PREPARE →

- ✓ Sanity Checks — Step type: parallel — 12 s
  - ✓ Sandbox variables — Step type: freestyle — 7 s
  - ✓ kubectl — Step type: skillshare/sanity-check-kubectl — 11 s
- ✓ Clone — Step type: parallel — 43 s
  - ✓ skillshare — Step type: git-clone — 43 s
  - ✓ skillshare-api — Step type: git-clone — 8 s
  - ✓ skillshare-web — Step type: git-clone — 8 s
  - ✓ api-gateway — Step type: git-clone — 7 s
  - ✓ membership — Step type: git-clone — 6 s
  - ✓ analytics-router — Step type: git-clone — 7 s
  - ✓ nginx-router — Step type: git-clone — 6 s
- ✓ Generate Manifests — Step type: parallel — 51 s
  - ✓ analytics-router — Step type: skillshare/helm — 49 s
  - ✓ skillshare-web — Step type: skillshare/helm — 50 s
  - ✓ skillshare-api — Step type: skillshare/helm — 50 s
  - ✓ skillshare — Step type: skillshare/helm — 51 s
  - ✓ nginx-router — Step type: skillshare/helm — 50 s
  - ✓ membership — Step type: skillshare/helm — 51 s
  - ✓ api-gateway — Step type: skillshare/helm — 49 s

PREFLIGHT →

- ✓ Kubeval — Step type: parallel — 27 s
  - ✓ analytics-router — Step type: skillshare/kubeval — 25 s
  - ✓ skillshare-web — Step type: skillshare/kubeval — 26 s
  - ✓ skillshare-api — Step type: skillshare/kubeval — 26 s
  - ✓ skillshare — Step type: skillshare/kubeval — 26 s
  - ✓ nginx-router — Step type: skillshare/kubeval — 27 s
  - ✓ membership — Step type: skillshare/kubeval — 26 s
  - ✓ api-gateway — Step type: skillshare/kubeval — 26 s
- ✓ Upload assets — Step type: skillshare/run-job — 2 min 42 s
- ✓ Preflight Checks — Step type: parallel — 1 min 41 s
  - ✓ analytics-router — Step type: skillshare/run-job — 40 s
  - ✓ skillshare-web — Step type: skillshare/run-job — 1 min 40 s
  - ✓ skillshare-api — Step type: skillshare/run-job — 1 min 17 s
  - ✓ skillshare — Step type: skillshare/run-job — 32 s
  - ✓ nginx-router — Step type: skillshare/run-job — 38 s
  - ✓ membership — Step type: skillshare/run-job — 42 s
  - ✓ api-gateway — Step type: skillshare/run-job — 51 s

DEPLOY →

- ✓ rolling_deploy — Step type: parallel — 3 min 42 s
  - ✓ skillshare — Step type: skillshare/rolling-deploy — 45 s
  - ✓ skillshare-api — Step type: skillshare/rolling-deploy — 1 min 0 s
  - ✓ skillshare-web — Step type: skillshare/rolling-deploy — 24 s
  - ✓ nginx-router — Step type: skillshare/rolling-deploy — 54 s
  - ✓ analytics-router — Step type: skillshare/rolling-deploy — 24 s
  - ✓ membership — Step type: skillshare/rolling-deploy — 20 s
  - ✓ api-gateway — Step type: skillshare/rolling-deploy — 3 min 41 s

VERIFY

- ✓ Smoke Test — Step type: parallel — 1 min 5 s
  - ✓ analytics-router — Step type: skillshare/run-job — 22 s
  - ✓ skillshare-web — Step type: skillshare/run-job — 24 s
  - ✓ skillshare-api — Step type: skillshare/run-job — 23 s
  - ✓ skillshare — Step type: skillshare/run-job — 1 min 4 s
  - ✓ nginx-router — Step type: skillshare/run-job — 22 s
  - ✓ membership — Step type: skillshare/run-job — 23 s
  - ✓ api-gateway — Step type: skillshare/run-job — 23 s

WRAP-UP

- ✓ Report Success — Step type: freestyle — 7 s
- Report failure — Step type: freestyle

Really had to zoom out for this one! 🔎

Success 🏁🏁

OK, let's rock!

1

Secrets with sops

👏 secrets managed in code

2

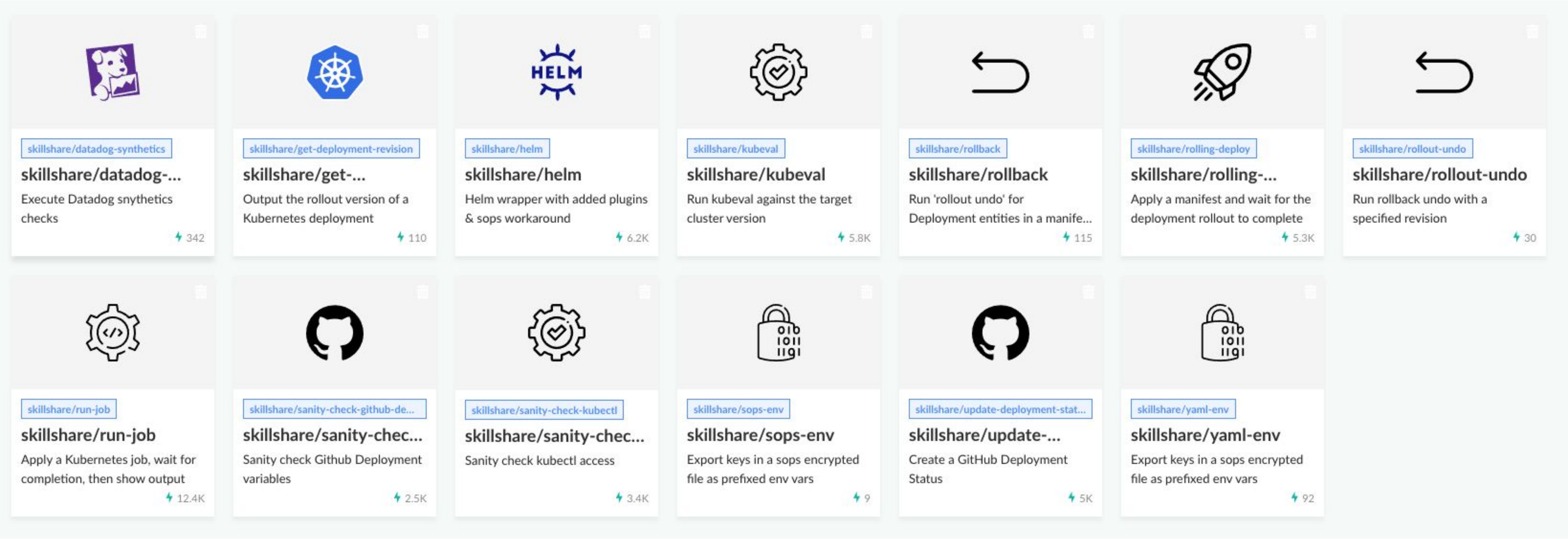
Multi-repo pipeline

Supports our sandbox

3

Extendable

Canaries, DataDog SLI verification, custom steps, we think; we can do it. 🚀



skillshare/datadog-synthetics

skillshare/datadog-...

Execute Datadog synthetics checks

⚡ 342

skillshare/get-deployment-revision

skillshare/get-...

Output the rollout version of a Kubernetes deployment

⚡ 110

skillshare/helm

skillshare/helm

Helm wrapper with added plugins & sops workaround

⚡ 6.2K

skillshare/kubeval

skillshare/kubeval

Run kubeval against the target cluster version

⚡ 5.8K

skillshare/rollback

skillshare/rollback

Run 'rollout undo' for Deployment entities in a manife...

⚡ 115

skillshare/rolling-deploy

skillshare/rolling-...

Apply a manifest and wait for the deployment rollout to complete

⚡ 5.3K

skillshare/rollout-undo

skillshare/rollout-undo

Run rollback undo with a specified revision

⚡ 30

skillshare/run-job

skillshare/run-job

Apply a Kubernetes job, wait for completion, then show output

⚡ 12.4K

skillshare/sanity-check-github-de...

skillshare/sanity-chec...

Sanity check Github Deployment variables

⚡ 2.5K

skillshare/sanity-check-kubectl

skillshare/sanity-chec...

Sanity check kubectl access

⚡ 3.4K

skillshare/sops-env

skillshare/sops-env

Export keys in a sops encrypted file as prefixed env vars

⚡ 9

skillshare/update-deployment-stat...

skillshare/update-...

Create a GitHub Deployment Status

⚡ 5K

skillshare/yaml-env

skillshare/yaml-env

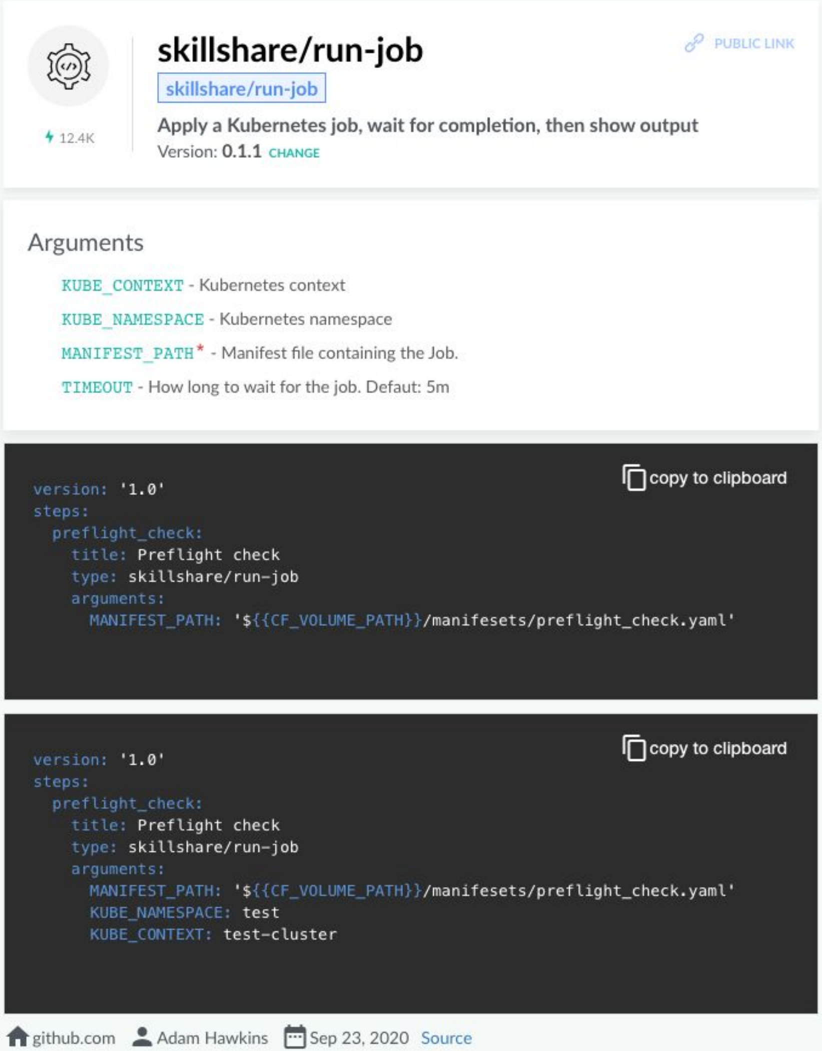
Export keys in a sops encrypted file as prefixed env vars

⚡ 92

The Skillshare Step Library

Lego blocks for developers to build pipelines

SKILL
SHare.



skillshare/run-job

skillshare/run-job

Apply a Kubernetes job, wait for completion, then show output

Version: **0.1.1** CHANGE

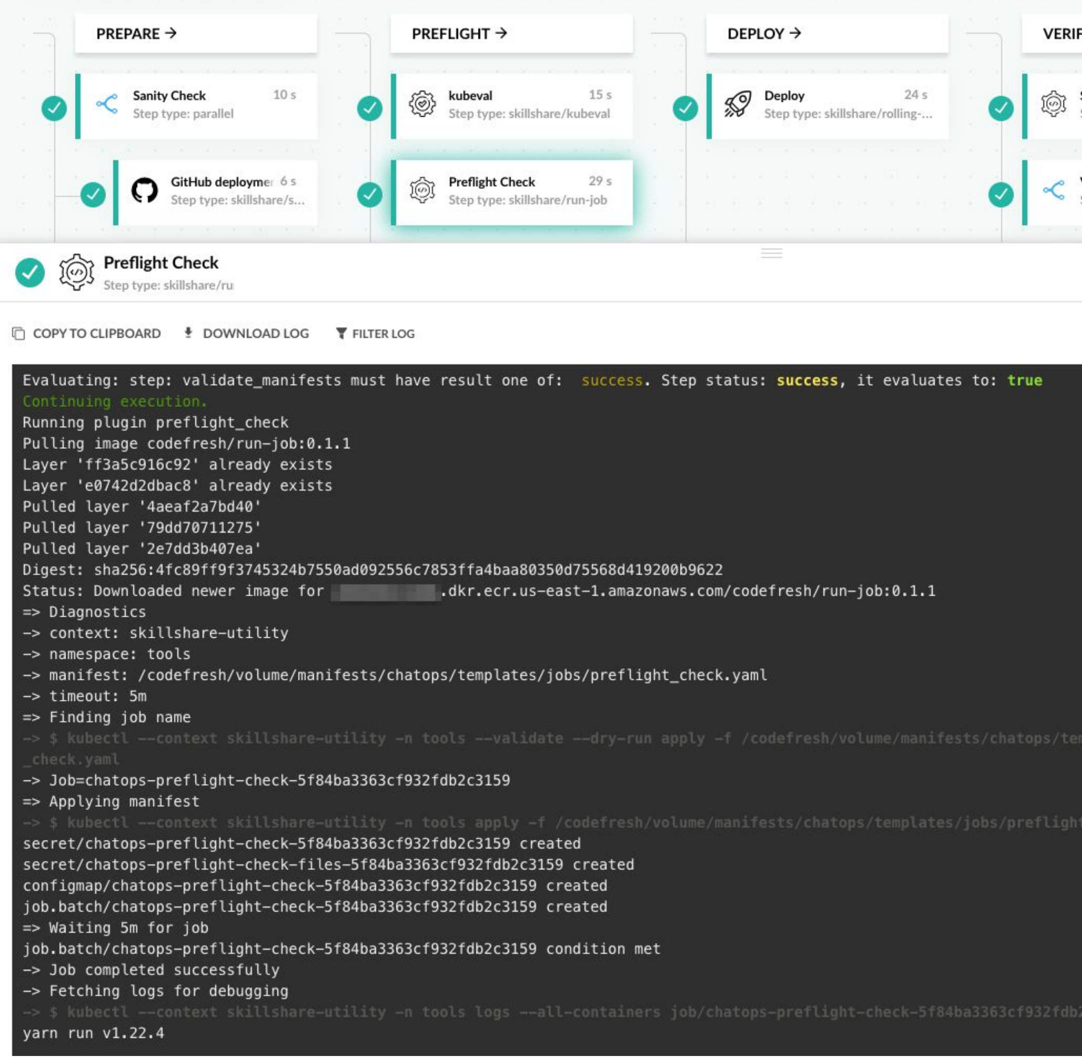
Arguments

- `KUBE_CONTEXT` - Kubernetes context
- `KUBE_NAMESPACE` - Kubernetes namespace
- `MANIFEST_PATH`* - Manifest file containing the Job.
- `TIMEOUT` - How long to wait for the job. Defaut: 5m

```yaml
version: '1.0'
steps:
  preflight_check:
    title: Preflight check
    type: skillshare/run-job
    arguments:
      MANIFEST_PATH: '${{CF_VOLUME_PATH}}/manifesets/preflight_check.yaml'
```

```yaml
version: '1.0'
steps:
  preflight_check:
    title: Preflight check
    type: skillshare/run-job
    arguments:
      MANIFEST_PATH: '${{CF_VOLUME_PATH}}/manifesets/preflight_check.yaml'
      KUBE_NAMESPACE: test
      KUBE_CONTEXT: test-cluster
```

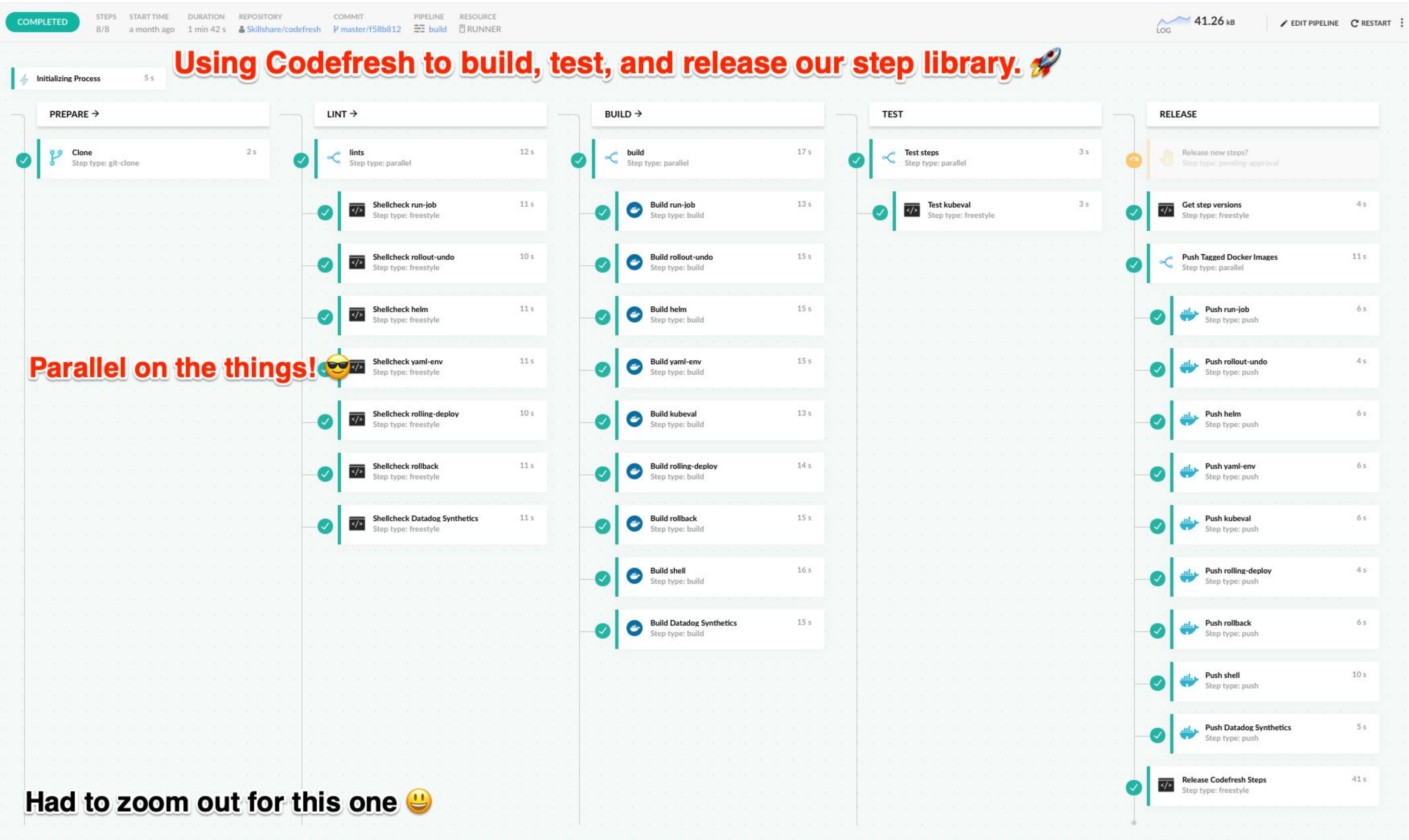
github.com Adam Hawkins Sep 23, 2020 Source



Preflight Check

Step type: skillshare/ru

```
Evaluating: step: validate_manifests must have result one of:  success. Step status: success, it evaluates to: true
Continuing execution.
Running plugin preflight_check
Pulling image codefresh/run-job:0.1.1
Layer 'ff3a5c916c92' already exists
Layer 'e0742d2dbac8' already exists
Pulled layer '4aeaf2a7bd40'
Pulled layer '79dd70711275'
Pulled layer '2e7dd3b407ea'
Digest: sha256:4fc89ff9f3745324b7550ad092556c7853ffa4baa80350d75568d419200b9622
Status: Downloaded newer image for [redacted].dkr.ecr.us-east-1.amazonaws.com/codefresh/run-job:0.1.1
=> Diagnostics
-> context: skillshare-utility
-> namespace: tools
-> manifest: /codefresh/volume/manifests/chatops/templates/jobs/preflight_check.yaml
-> timeout: 5m
=> Finding job name
-> $ kubectl --context skillshare-utility -n tools --validate --dry-run apply -f /codefresh/volume/manifests/chatops/te
_check.yaml
-> Job=chatops-preflight-check-5f84ba3363cf932fdb2c3159
=> Applying manifest
-> $ kubectl --context skillshare-utility -n tools apply -f /codefresh/volume/manifests/chatops/templates/jobs/prefligh
secret/chatops-preflight-check-5f84ba3363cf932fdb2c3159 created
secret/chatops-preflight-check-files-5f84ba3363cf932fdb2c3159 created
configmap/chatops-preflight-check-5f84ba3363cf932fdb2c3159 created
job.batch/chatops-preflight-check-5f84ba3363cf932fdb2c3159 created
=> Waiting 5m for job
job.batch/chatops-preflight-check-5f84ba3363cf932fdb2c3159 condition met
-> Job completed successfully
-> Fetching logs for debugging
-> $ kubectl --context skillshare-utility -n tools logs --all-containers job/chatops-preflight-check-5f84ba3363cf932fdb
yarn run v1.22.4
```



Using Codefresh to build, test, and release our step library. 🚀

Parallel on the things! 😎

Had to zoom out for this one 😃

COMPLETED

STEPS 8/8 · START TIME a month ago · DURATION 1 min 42 s · REPOSITORY Skillshare/codefresh · COMMIT master/f58b812 · PIPELINE build · RESOURCE RUNNER

LOG 41.26 kB · EDIT PIPELINE · RESTART

Initializing Process 5 s

PREPARE →

- Clone 2 s — Step type: git-clone

LINT →

- lints 12 s — Step type: parallel
  - Shellcheck run-job 11 s — Step type: freestyle
  - Shellcheck rollout-undo 10 s — Step type: freestyle
  - Shellcheck helm 11 s — Step type: freestyle
  - Shellcheck yaml-env 11 s — Step type: freestyle
  - Shellcheck rolling-deploy 10 s — Step type: freestyle
  - Shellcheck rollback 11 s — Step type: freestyle
  - Shellcheck Datadog Synthetics 11 s — Step type: freestyle

BUILD →

- build 17 s — Step type: parallel
  - Build run-job 13 s — Step type: build
  - Build rollout-undo 15 s — Step type: build
  - Build helm 15 s — Step type: build
  - Build yaml-env 15 s — Step type: build
  - Build kubeval 13 s — Step type: build
  - Build rolling-deploy 14 s — Step type: build
  - Build rollback 15 s — Step type: build
  - Build shell 16 s — Step type: build
  - Build Datadog Synthetics 15 s — Step type: build

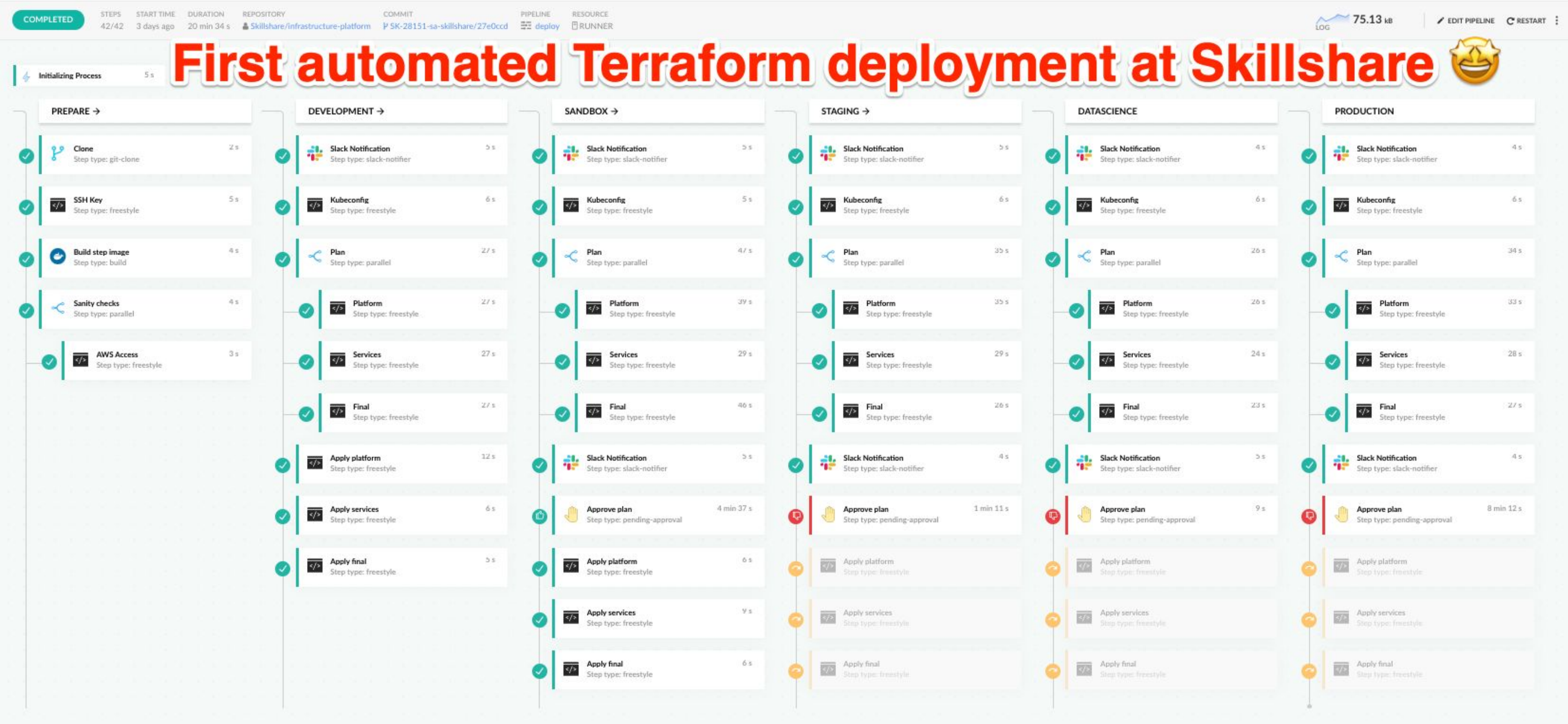
TEST

- Test steps 3 s — Step type: parallel
  - Test kubeval 3 s — Step type: freestyle

RELEASE

- Release new steps? — Step type: pending-approval
- Get step versions 4 s — Step type: freestyle
- Push Tagged Docker Images 11 s — Step type: parallel
  - Push run-job 6 s — Step type: push
  - Push rollout-undo 4 s — Step type: push
  - Push helm 6 s — Step type: push
  - Push yaml-env 6 s — Step type: push
  - Push kubeval 6 s — Step type: push
  - Push rolling-deploy 4 s — Step type: push
  - Push rollback 6 s — Step type: push
  - Push shell 10 s — Step type: push
  - Push Datadog Synthetics 5 s — Step type: push
- Release Codefresh Steps 41 s — Step type: freestyle

First automated Terraform deployment at Skillshare 🤩



COMPLETED

STEPS 42/42 · START TIME 3 days ago · DURATION 20 min 34 s · REPOSITORY Skillshare/infrastructure-platform · COMMIT SK-28151-sa-skillshare/27e0ccd · PIPELINE deploy · RESOURCE RUNNER

LOG 75.13 kB · EDIT PIPELINE · RESTART

Initializing Process 5 s

PREPARE →

- Clone — Step type: git-clone — 2 s
- SSH Key — Step type: freestyle — 5 s
- Build step image — Step type: build — 4 s
- Sanity checks — Step type: parallel — 4 s
  - AWS Access — Step type: freestyle — 3 s

DEVELOPMENT →

- Slack Notification — Step type: slack-notifier — 5 s
- Kubeconfig — Step type: freestyle — 6 s
- Plan — Step type: parallel — 27 s
  - Platform — Step type: freestyle — 27 s
  - Services — Step type: freestyle — 27 s
  - Final — Step type: freestyle — 27 s
- Apply platform — Step type: freestyle — 12 s
- Apply services — Step type: freestyle — 6 s
- Apply final — Step type: freestyle — 5 s

SANDBOX →

- Slack Notification — Step type: slack-notifier — 5 s
- Kubeconfig — Step type: freestyle — 5 s
- Plan — Step type: parallel — 47 s
  - Platform — Step type: freestyle — 39 s
  - Services — Step type: freestyle — 29 s
  - Final — Step type: freestyle — 46 s
- Slack Notification — Step type: slack-notifier — 5 s
- Approve plan — Step type: pending-approval — 4 min 37 s
- Apply platform — Step type: freestyle — 6 s
- Apply services — Step type: freestyle — 9 s
- Apply final — Step type: freestyle — 6 s

STAGING →

- Slack Notification — Step type: slack-notifier — 5 s
- Kubeconfig — Step type: freestyle — 6 s
- Plan — Step type: parallel — 35 s
  - Platform — Step type: freestyle — 35 s
  - Services — Step type: freestyle — 29 s
  - Final — Step type: freestyle — 26 s
- Slack Notification — Step type: slack-notifier — 4 s
- Approve plan — Step type: pending-approval — 1 min 11 s
- Apply platform — Step type: freestyle
- Apply services — Step type: freestyle
- Apply final — Step type: freestyle

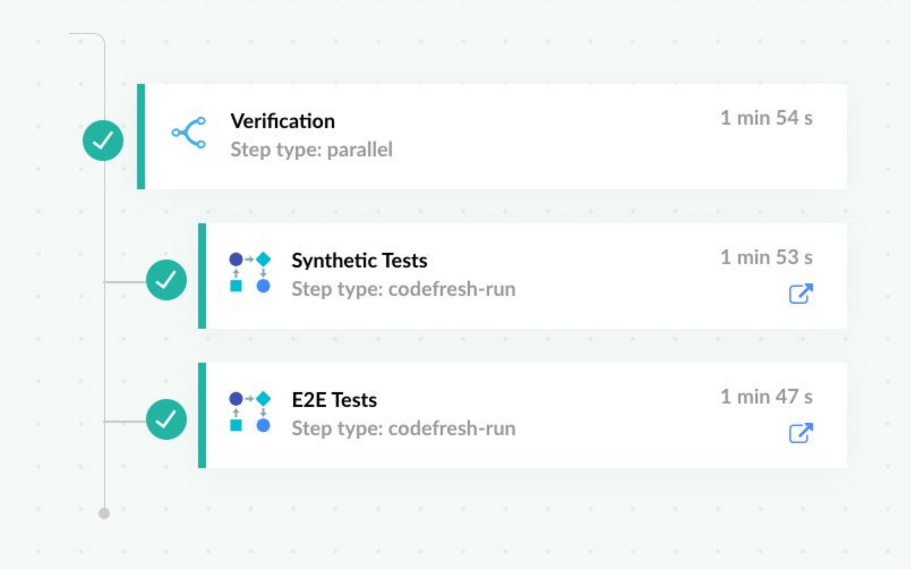
DATASCIENCE

- Slack Notification — Step type: slack-notifier — 4 s
- Kubeconfig — Step type: freestyle — 6 s
- Plan — Step type: parallel — 26 s
  - Platform — Step type: freestyle — 26 s
  - Services — Step type: freestyle — 24 s
  - Final — Step type: freestyle — 23 s
- Slack Notification — Step type: slack-notifier — 5 s
- Approve plan — Step type: pending-approval — 9 s
- Apply platform — Step type: freestyle
- Apply services — Step type: freestyle
- Apply final — Step type: freestyle

PRODUCTION

- Slack Notification — Step type: slack-notifier — 4 s
- Kubeconfig — Step type: freestyle — 6 s
- Plan — Step type: parallel — 34 s
  - Platform — Step type: freestyle — 33 s
  - Services — Step type: freestyle — 28 s
  - Final — Step type: freestyle — 27 s
- Slack Notification — Step type: slack-notifier — 4 s
- Approve plan — Step type: pending-approval — 8 min 12 s
- Apply platform — Step type: freestyle
- Apply services — Step type: freestyle
- Apply final — Step type: freestyle

Composable Pipelines



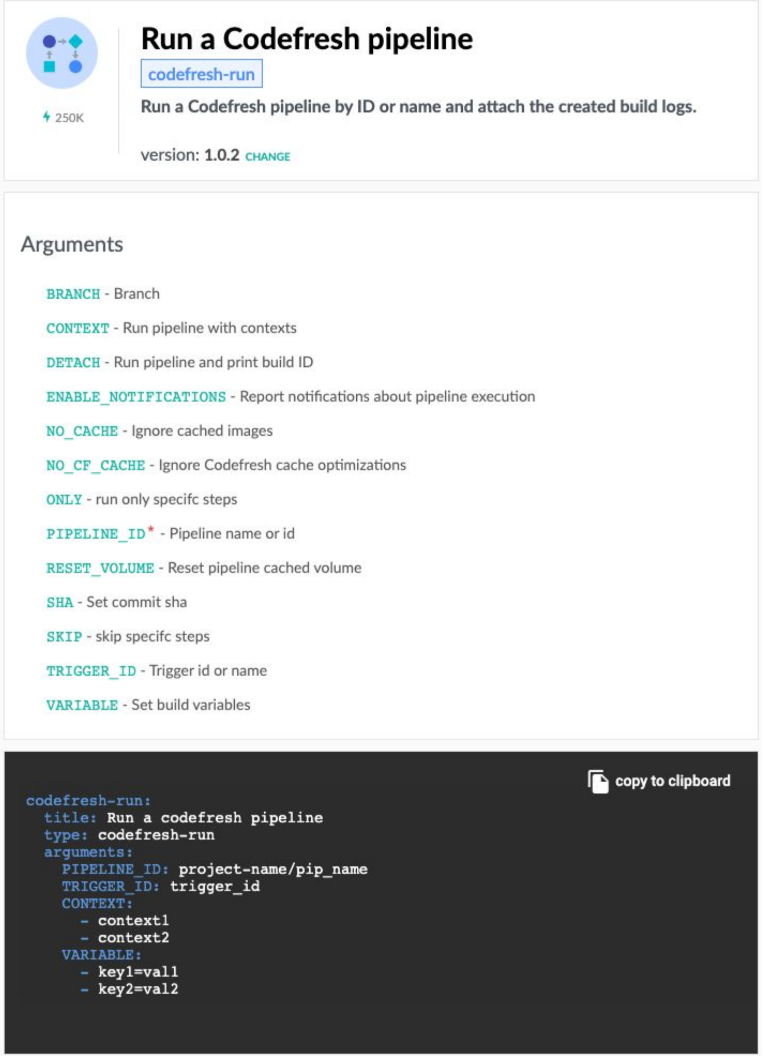
Verification
Step type: parallel
1 min 54 s

Synthetic Tests
Step type: codefresh-run
1 min 53 s

E2E Tests
Step type: codefresh-run
1 min 47 s

Use pipelines as team APIs. Want e2e tests in our pipeline? Go ahead and add it.

skill share.



Run a Codefresh pipeline

Run a Codefresh pipeline by ID or name and attach the created build logs.

version: 1.0.2 CHANGE

Arguments

- BRANCH - Branch
- CONTEXT - Run pipeline with contexts
- DETACH - Run pipeline and print build ID
- ENABLE_NOTIFICATIONS - Report notifications about pipeline execution
- NO_CACHE - Ignore cached images
- NO_CF_CACHE - Ignore Codefresh cache optimizations
- ONLY - run only specifc steps
- PIPELINE_ID* - Pipeline name or id
- RESET_VOLUME - Reset pipeline cached volume
- SHA - Set commit sha
- SKIP - skip specifc steps
- TRIGGER_ID - Trigger id or name
- VARIABLE - Set build variables

```
codefresh-run:
  title: Run a codefresh pipeline
  type: codefresh-run
  arguments:
    PIPELINE_ID: project-name/pip_name
    TRIGGER_ID: trigger_id
    CONTEXT:
      - context1
      - context2
    VARIABLE:
      - key1=val1
      - key2=val2
```

Code Samples

Q&A