Servicios de CI/CD

Circle CI

Set up new projects or follow projects already building on CircleCI.
Following a project adds it to your dashboard.

Projects

Repo name

Follow All

Repo

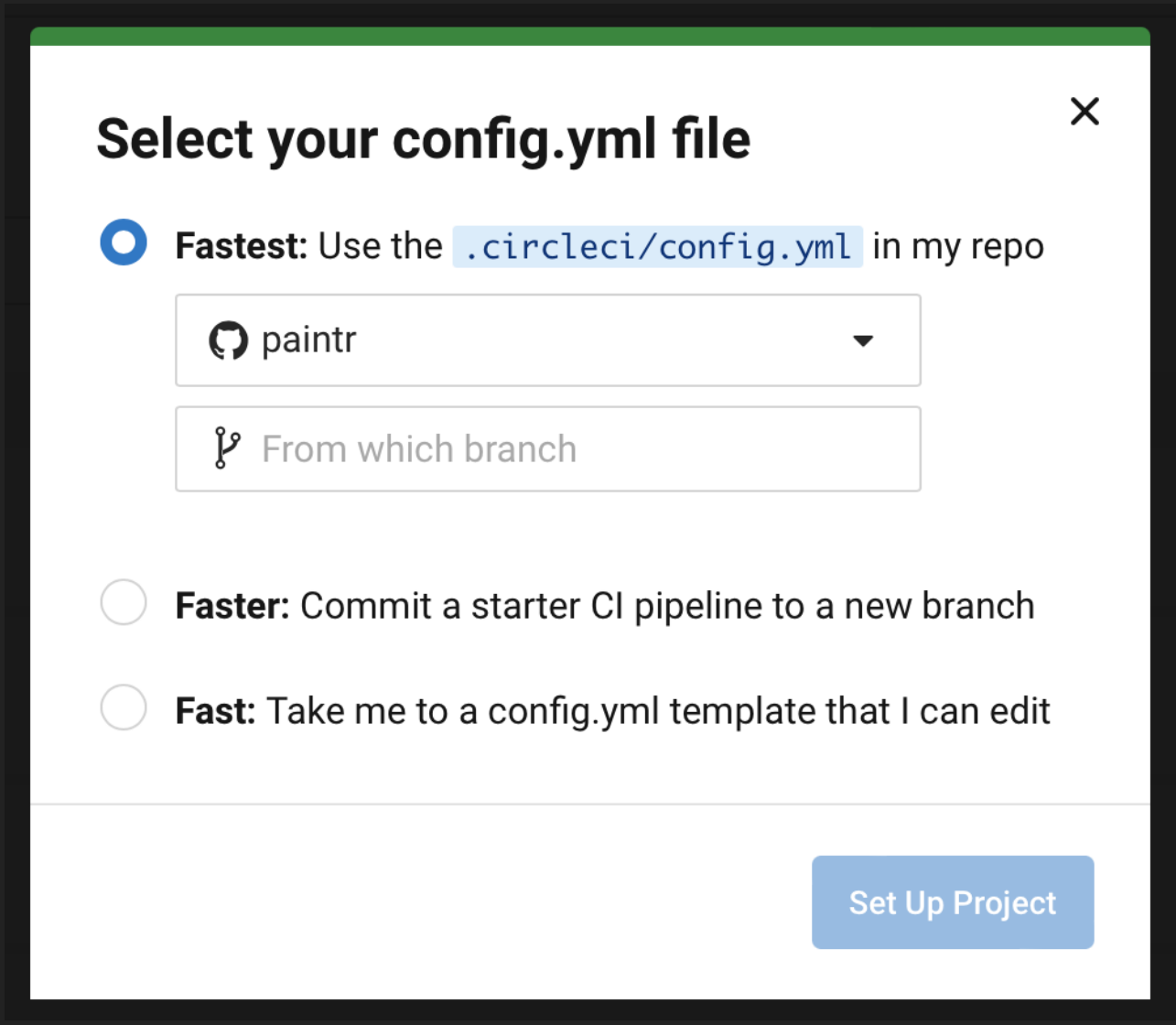
paintr	Set Up Project
UMA	Set Up Project
ubay	Set Up Project
Demeter	Set Up Project
expensive-dolphin	Set Up Project
minio-java	Set Up Project
rquote	Set Up Project
lab2-mantenimiento	Set Up Project
Programming-Notes	Set Up Project

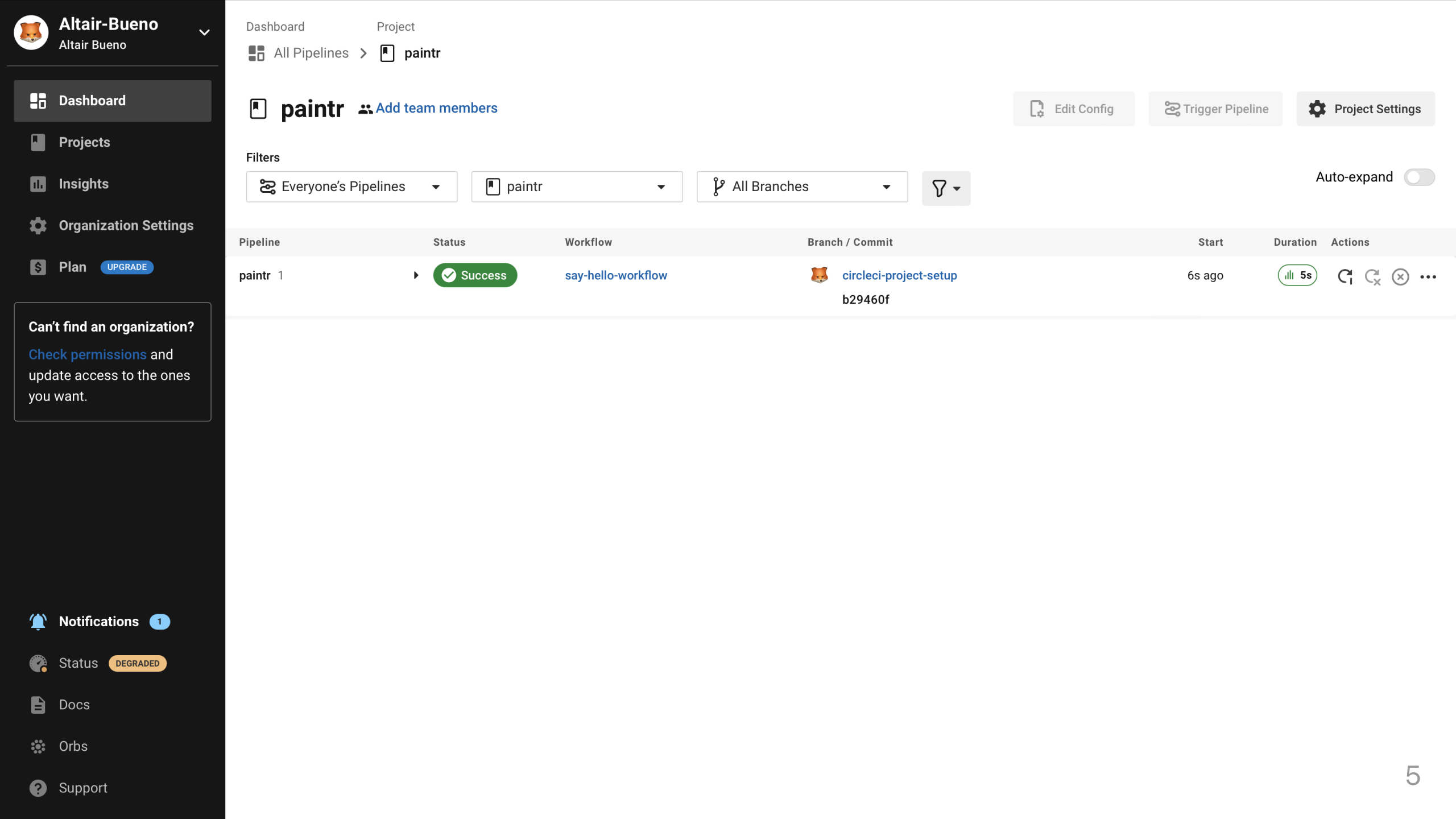
Learn

Popular docs

Setting up new projects
Following projects
Configuration reference
Using the CircleCI CLI

- **Fastest**: Utiliza un fichero ya existente en el repo
- **Faster**: Crea una rama `circleci-project-setup` con un fichero `.circleci/config.yml`
- **Fast**: Te permite configurar el fichero `.circleci/config.yml` de forma
  similar a GitHub Actions




Altair-Bueno
Altair Bueno
Dashboard
Projects
Insights
Organization Settings
Plan
UPGRADE
Can't find an organization?
Check permissions and update access to the ones you want.
Notifications
1
Status
DEGRADED
Docs
Orbs
Support
Dashboard
Project
All Pipelines
paintr
paintr
Add team members
Edit Config
Trigger Pipeline
Project Settings
Filters
Everyone's Pipelines
paintr
All Branches
Auto-expand
Pipeline
Status
Workflow
Branch / Commit
Start
Duration
Actions
paintr 1
Success
say-hello-workflow
circleci-project-setup
b29460f
6s ago
5s

Altair-Bueno
Altair Bueno

- Dashboard
- Projects
- Insights
- Organization Settings
- Plan UPGRADE

Can't find an organization?

Check permissions and update access to the ones you want.

- Notifications 1
- Status DEGRADED
- Docs
- Orbs
- Support

Dashboard > Project > Branch > Workflow

All Pipelines > paintr > circleci-project-setup > say-hello-workflow

say-hello-workflow

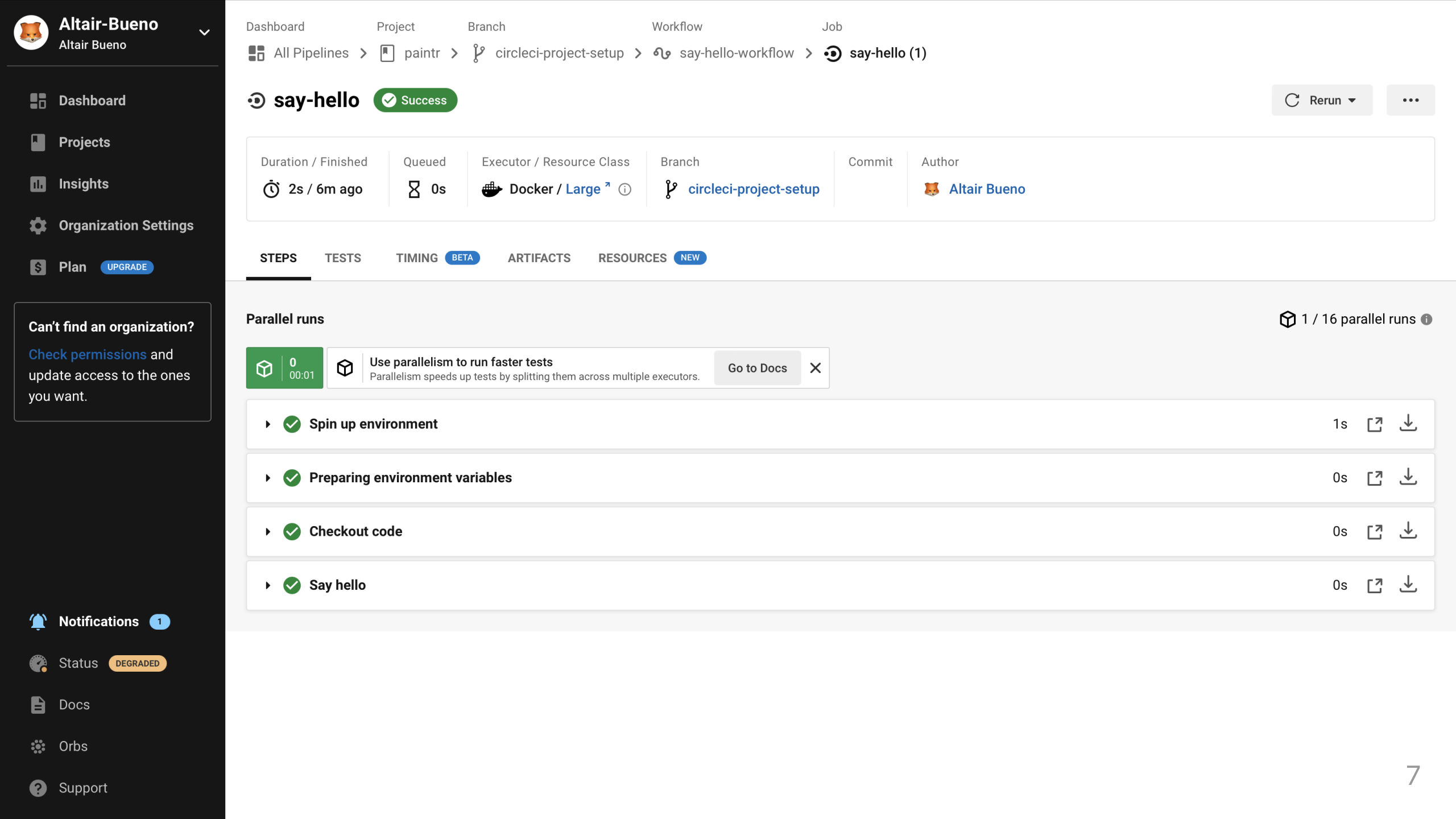
Success

Insights | Rerun | ...

Duration / Finished	Branch	Commit	Author
5s / 3m ago	circleci-project-setup	b29460f	Altair-Bueno

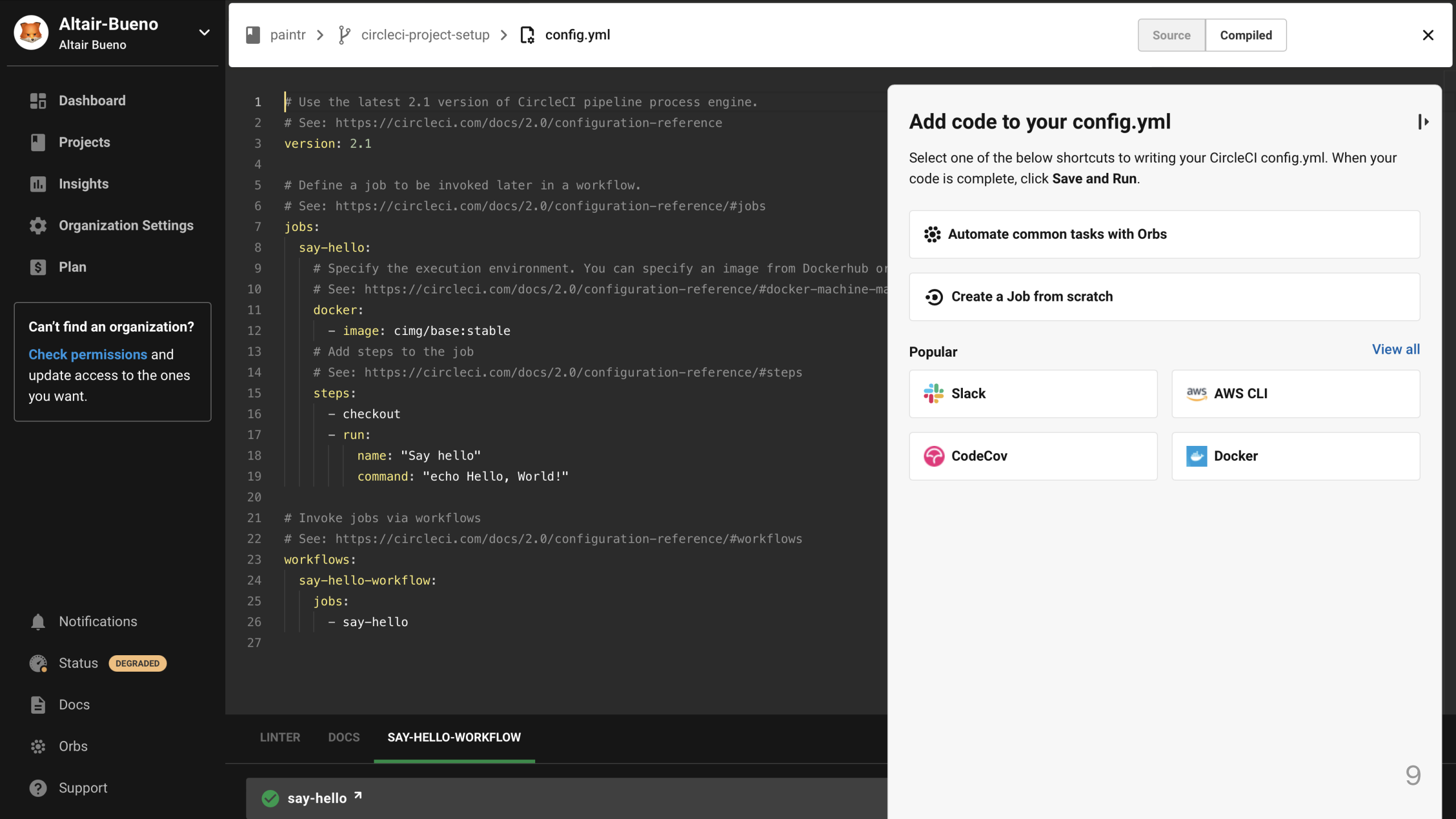
say-hello 3s

Did you know? CircleCI teams that commit 4x as often fix failed builds 2x faster. Learn more


Altair-Bueno
Altair Bueno
Dashboard
Projects
Insights
Organization Settings
Plan
UPGRADE
Can't find an organization?
Check permissions and update access to the ones you want.
Notifications
1
Status
DEGRADED
Docs
Orbs
Support
Dashboard
Project
Branch
Workflow
Job
All Pipelines
paintr
circleci-project-setup
say-hello-workflow
say-hello (1)
say-hello
Success
Rerun
Duration / Finished
2s / 6m ago
Queued
0s
Executor / Resource Class
Docker / Large
Branch
circleci-project-setup
Commit
Author
Altair Bueno
STEPS
TESTS
TIMING
BETA
ARTIFACTS
RESOURCES
NEW
Parallel runs
1 / 16 parallel runs
0
00:01
Use parallelism to run faster tests
Parallelism speeds up tests by splitting them across multiple executors.
Go to Docs
Spin up environment
1s
Preparing environment variables
0s
Checkout code
0s
Say hello
0s

`config.yml`

Fichero YAML ubicado en la carpeta `.circleci` en cualquier rama del repositorio. Podemos editarlo manualmente desde nuestro editor de código favorito o desde la web


Altair-Bueno
Altair Bueno
Dashboard
Projects
Insights
Organization Settings
Plan
Can't find an organization?
Check permissions and update access to the ones you want.
Notifications
Status DEGRADED
Docs
Orbs
Support
paintr > circleci-project-setup > config.yml
Source
Compiled
1 # Use the latest 2.1 version of CircleCI pipeline process engine.
2 # See: https://circleci.com/docs/2.0/configuration-reference
3 version: 2.1
4
5 # Define a job to be invoked later in a workflow.
6 # See: https://circleci.com/docs/2.0/configuration-reference/#jobs
7 jobs:
8 say-hello:
9 # Specify the execution environment. You can specify an image from Dockerhub or
10 # See: https://circleci.com/docs/2.0/configuration-reference/#docker-machine-ma
11 docker:
12 - image: cimg/base:stable
13 # Add steps to the job
14 # See: https://circleci.com/docs/2.0/configuration-reference/#steps
15 steps:
16 - checkout
17 - run:
18 name: "Say hello"
19 command: "echo Hello, World!"
20
21 # Invoke jobs via workflows
22 # See: https://circleci.com/docs/2.0/configuration-reference/#workflows
23 workflows:
24 say-hello-workflow:
25 jobs:
26 - say-hello
27
LINTER
DOCS
SAY-HELLO-WORKFLOW
say-hello
Add code to your config.yml
Select one of the below shortcuts to writing your CircleCI config.yml. When your code is complete, click Save and Run.
Automate common tasks with Orbs
Create a Job from scratch
Popular
View all
Slack
AWS CLI
CodeCov
Docker
9

Estructura

```yaml
version: 2.1
# Orbs requeridos
orbs:
  node: circleci/node@4.7.0
jobs:
  build:
    # Entorno de ejecución (`docker`, `machine`, `macos` or `executor`)
    executor:
      name: node/default
      tag: "10.4"
    # Etapas que componen el trabajo
    steps:
      - checkout
      - node/with-cache:
          steps:
            - run: npm install
      - run: npm run test
```

Orbs

- Utilidades por la comunidad o por Circle CI
- Funcionan como **módulos**
- Se importan y se exportan
- Simplifican enormemente el desarrollo de workflows multiplataforma

```yaml
version: "2.1"
orbs:
  # Orb de Rust (aka. `from circleci import rust@x.y.z as rust `)
  rust: circleci/rust@x.y.z
workflows:
  # Definimos los distintos workflows disponibles (aka. `production.py`)
  production:
    jobs:
      # Ejecutamos la función `build` dentro del orb definido
      # anteriormente (aka. `rust.build(release=True)`)
      - rust/build:
          release: true
```

Fuente: https://circleci.com/developer/orbs/orb/circleci/rust#commands-build

Jobs

- Funciones a llamar para ser ejecutadas
- Cada job puede ser en un entorno distinto
- Pueden definirse globalmente para todos los workflows o para un workflow concreto

```yaml
version: 2
jobs: # we now have TWO jobs, so that a workflow can coordinate them!
  # aka. `def hello-world():`
  hello-world: # This is our first job.
    docker: # it uses the docker executor
      - image: cimg/ruby:2.6.8 # specifically, a docker image with ruby 2.6.8
        auth:
          username: mydockerhub-user
          password: $DOCKERHUB_PASSWORD # context / project UI env-var reference
    # Steps are a list of commands to run inside the docker container above.
    steps:
      - checkout # this pulls code down from GitHub
      # aka. `hello-world()`
      - run: echo "A first hello" # This prints "A first hello" to stdout.
workflows:
  version: 2
  one_and_two: # this is the name of our workflow
    jobs: # and here we list the jobs we are going to run.
      - hello-world
```

Demo

Link: Altair-Bueno/paintr

```yaml
version: 2.1
executors:
  rust-executor:
    docker:
      - image: cimg/rust:1.60.0
jobs:
  test-kahlo:
    executor: rust-executor
    steps:
      - checkout
      - restore_cache:
          key: kahlo-rust-cache-{{ checksum "kahlo/Cargo.lock" }}
      - run: |
          cd ./kahlo
          cargo test
      - save_cache:
          key: kahlo-rust-cache-{{ checksum "kahlo/Cargo.lock" }}
          paths:
            - "~/.cargo"
            - "./kahlo/target"
orbs:
  node: circleci/node@5.0.2
workflows:
  test:
    jobs:
      - test-kahlo
      - node/run:
          app-dir: "./dali"
          npm-run: "test"
```

Planes y precios

Fuente: https://circleci.com/pricing/

START HERE

CLOUD

Free

Do more with your minutes.
Give your team the best CI/CD.

$0 per month

Start for Free

- Up to 6,000 build minutes per month
- **Largest selection:** Build for Docker, Windows, Linux, Arm, and macOS or on your own compute with self-hosted runners
- **Build better:** Choose the right resource class size (S–L) to go fast and maximize your build minutes
- **Fast:** Run up to 30 jobs at a time and run your tests in parallel using test splitting

CLOUD

Performance

Pay for what you use. Build with the flexibility your business requires.

Starting at

$15 per month

Get Started

- 6,000 bonus build minutes per month, pay as you go for more
- **Teamwork:** 5 user seats included, add more as you grow
- **Power:** All of the environments in Free, plus access to larger resource classes for better performance
- **Speed:** Run up to 80 jobs concurrently
- **Expertise:** Access to 8x5 support with add-on plan

CLOUD

Scale

Enterprise-level confidence and support for teams who deliver.

Starting at

$2,000 per month

Contact Us

- Customize your build minutes and seats to fit your team
- **Power:** Our largest resource classes for complex processes and speed
- **Flexibility:** All the environments in Performance, plus access to GPU resource classes
- **Expertise:** A dedicated account team and access to 24/7 support
- **Control:** Customizable annual billing, audit logging, and bulk data export

SELF-HOSTED

Server

The power of CircleCI, on-prem or in your private cloud.

Custom

Contact Us

Learn more

- **Control:** CircleCI behind your firewall or on your own hardware
- **Scale:** Unlimited build minutes, 30 user seats included, add more as you grow
- **Flexibility:** Build on Linux, Windows, Arm, and Android
- **Expertise:** A dedicated account team and access to 24/7 support, including migration assistance

Prices do not include applicable taxes.

	Hardware*	Creditos	Usuarios	Precio	Público objetivo
Free **	VMs	30.000	1	Gratis	Desarrolladores
Performance	VMs + macOS	55.000	5 gratuitos	Desde $15	Pequeñas empresas
Scale	Todo	Custom	Custom	$2.000	Pequeñas/medianas empresas
Server	Todo	Custom	Custom	Custom	Grandes empresas

> * Docker, VMs (linux, windows, macOS, ARM linux), baremetal macOS, NVIDIA GPU cluster (versiones linux y windows)
>
> ** Proyectos OSS tienen más créditos

Créditos

Moneda digital de Circle CI utilizado para pagar, en tiempo real, por los servicios utilizados

Servicios que consumen creditos

- Cada runner tiene un precio de **créditos/minuto** (warm-up time no cuenta)
- **Número de usuarios activos/mes**
- **Red y almacenamiento**: Direcciones IP, paquetes de red y almacenamiento que excedan la cuota
- **Miscelanea**: Las funciones extra también se pagan mediante créditos

Importante!!

- Los créditos que pertenecen a tu plan que no gastes desaparecerán al mes siguiente
- Si tus créditos bajan del 2%, Circle CI automáticamente renovará un 25% de ellos
- Si cancelas tu cuenta, no recuperas el dinero. D:

Puntos fuertes

- Permite configurar runners concurrentes de forma sencilla
- Amplio abanico de ofertas hardware. Especialmente interesante el soporte para **gráficas NVIDIA** tanto en windows como linux
- Soporte de primer grado para contenedores de **Docker**
- El plan gratuito no requiere tarjeta de crédito
- Acceso a los workflows mediante ssh tras su finalización
- Todos los jobs se ejecutan en paralelo
- Los jobs no se ejecutan si los ficheros no se han cambiado

Conclusión

- **Circle CI**: Desarrollo multiplataforma, GPU computing o con alta dependencia en Dockers