



# CI2: Circle Boogaloo

Collin Brittle, Chris Colvard, David Cliff  
Samvera Connect 2019

# What is Continuous Integration (CI)?

- **Testing** - Run your test suite every time you commit, or open a PR
- **Linting** - Automatically enforce your style guide
- **Coverage** - Ensure your new code has tests
- **Automated** - Even if you forget to run your tests, CI won't.
- **Independent** - Avoids “works on my machine” syndrome

# Why CircleCI?

- **Free base tier for open source** - Samvera pays for more, but Circle gives you four cores for free, if your repo is public.
- **Parallelizable builds** - Circle can split up your tests and run multiple at the same time.
- **Works well with Docker** - And Docker allows us to reuse infrastructure components, which speeds up CI.
- **Rerun with SSH** - If you want more insight into your failing build, you can rerun it, and get access to the build server when it fails.

# CircleCI Basics

**Executor** - the environment that job steps will run; usually a set of docker containers

**Step** - an executable command; predefined commands include checkout, save\_cache, restore\_cache, persist\_to\_workspace, attach\_workspace, and run which allows for custom shell command

**Job** - a collection of steps

**Workflow** - a defined run order for a set of jobs

# Simple config.yml

```
version: 2
jobs:
  build:
    docker:
      - image: circleci/ruby:2.4.1
    steps:
      - checkout
      - run: echo "A first hello"
```

# Workflow Example

```
jobs:
  build:
    docker:
      - image: circleci/<language>:<version TAG>
    steps:
      - checkout
      - run: <command>
  test:
    docker:
      - image: circleci/<language>:<version TAG>
    steps:
      - checkout
      - run: <command>
workflows:
  version: 2
  build_and_test:
    jobs:
      - build
      - test
```

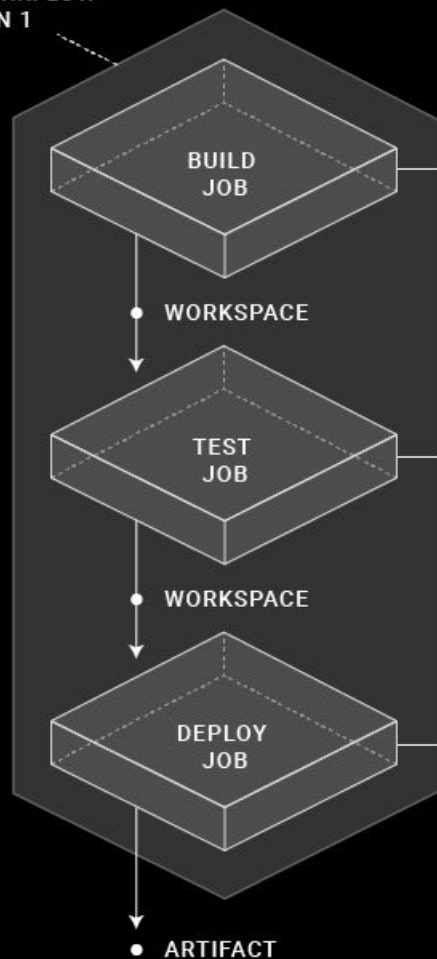
# CircleCI Persistence Features

**Caches** - persists data between the same job across different workflow builds

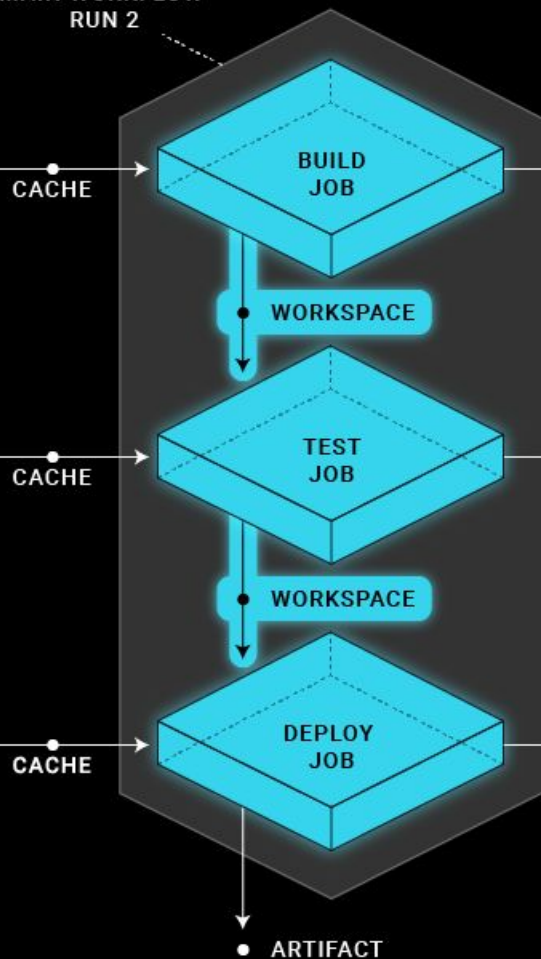
**Workspaces** - persists data between jobs within the same workflow build

**Artifacts** - persists data after a workflow has finished; think test and coverage reports

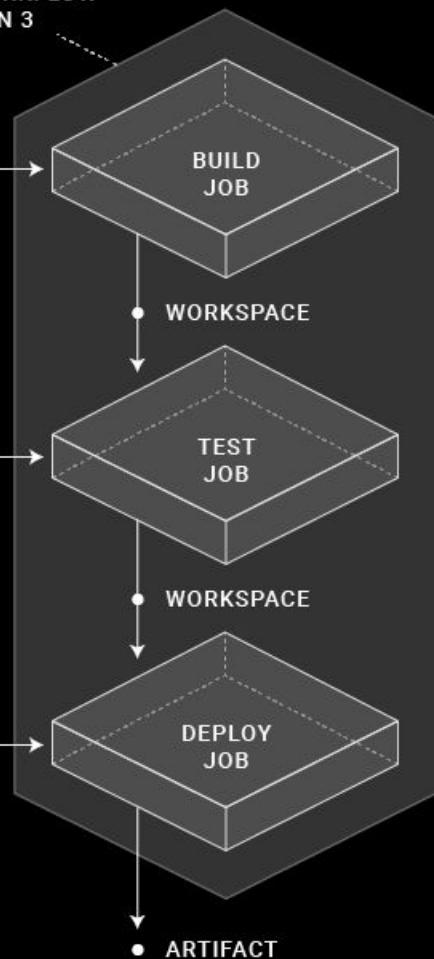
MAIN WORKFLOW  
RUN 1



MAIN WORKFLOW  
RUN 2



MAIN WORKFLOW  
RUN 3





# Caching Example

```
steps:
  - restore_cache:
      keys:
        - source-v1-{{ .Branch }}-{{ .Revision }}
        - source-v1-{{ .Branch }}-
        - source-v1-

  - checkout

  - save_cache:
      key: source-v1-{{ .Branch }}-{{ .Revision }}
      paths:
        - ".git"
```

# Workspace Example

```
build:
  steps:
    - run: echo "Hello World" >> persisted_file
    - persist to_workspace:
        root: ~/
        paths:
          - project/*

test:
  steps:
    - attach workspace:
        at: ~/
    - run: cat ~/project/persisted_file
```

# Artifacts Example

- run:
  - name: Build the Jekyll site
  - command: bundle exec jekyll build --source jekyll --destination jekyll/\_site/docs/
- store\_artifacts:
  - path: jekyll/\_site/docs/
  - destination: circleci-docs
- store\_test\_results:
  - path: test-results

Type	Lifetime	Use	Example
Artifacts	Months	Preserve long-term artifacts.	Available in the Artifacts tab of the Job page under the <code>tmp/circle-artifacts.&lt;hash&gt;/container</code> or similar directory.
Workspaces	Duration of workflow	Attach the workspace in a downstream container with the <code>attach_workspace: step</code> .	The <code>attach_workspace</code> copies and re-creates the entire workspace content when it runs.
Caches	Months	Store non-vital data that may help the job run faster, for example npm or Gem packages.	The <code>save_cache</code> job step with a path to a list of directories to add and a key to uniquely identify the cache (for example, the branch, build number, or revision). Restore the cache with <code>restore_cache</code> and the appropriate key.

# CircleCI Orbs

## What is an orb?

Collection of configuration blocks that can be shared and reused

Executors, Commands, and Jobs

## When is an orb helpful?

When many projects need the same configuration (like Samvera gems and apps) orbs keep the configuration simple and DRY. This also allows for mass updates when a common config needs changing (think headless chrome driver).

# The Samvera Orb

<https://circleci.com/orbs/registry/orb/samvera/circleci-orb>

<https://github.com/samvera-labs/samvera-circleci-orb>

# The Samvera Orb - Executors

## **ruby**

Uses circleci's ruby images (configurable with ruby\_version and ruby\_type parameters)

## **ruby\_fcrepo\_solr**

Same as above for ruby

University of Alberta's fcrepo image (configurable with fcrepo\_version)

Official Docker solr image (solr:7-alpine) with configurable port (solr\_port)

## **ruby\_fcrepo\_solr\_redis**

Same as above for ruby, fcrepo, and solr

CircleCI's redis image (configurable with redis\_version)

## **ruby\_fcrepo\_solr\_redis\_postgres**

Same as above for ruby, fcrepo, solr, and redis

CircleCI's postgres image (postgres:9.6.2-alpine)

# The Samvera Orb - Commands

## **cached\_checkout**

### **bundle / bundle\_for\_gem**

Installs specified version of bundler and caches bundle install

Uses ruby\_version, Gemfile and/or gemspec for cache key

### **engine\_cart\_generate**

Caches internal test app using cache\_key passed in

### **install\_solr\_core / install\_solr\_active\_fedora\_core**

Installs solr configuration from active\_fedora or specified solr\_config\_path

### **rubocop**

### **parallel\_rspec**

Runs rspec in parallelized containers with tests split by historical timings

Saves results (store\_test\_results and store\_artifacts)



# Simple Usage of Orb - iiif\_manifest

[https://github.com/samvera/iiif\\_manifest/blob/master/.circleci/config.yml](https://github.com/samvera/iiif_manifest/blob/master/.circleci/config.yml)

Executor: ruby

Steps:

- cached\_checkout
- bundle\_for\_gem
- rubocop
- parallel\_rspec

Build matrix: Ruby 2.4, 2.5, 2.6 and Rails 5.1, 5.2

# Complex Usage of Orb - Hyrax

<https://github.com/samvera/hyrax/blob/master/.circleci/config.yml>

Executor: ruby and ruby\_fcrepo\_solr\_redis

Jobs:

- Bundle
- Lint
- Build
- Test

Build matrix: Ruby 2.4, 2.5, 2.6 and Rails 5.2 (and Valkyrie)

# Using CircleCI in a New Samvera App

# Migrating to CircleCI

Gotchas, Pitfalls, Wish I Had Known

# Tips and Tricks

Restart workflow from failed

Restart job with SSH

Validating config file (circleci config validate)

Questions?

# Workshop

<https://gist.github.com/cjcolvar/191087ab5081e457d13c60f493a40cea>