

exclusive corporation

Sidekiq::Portal

https://github.com/0exp/sidekiq_portal

Sidekiq::Portal - Что удалось сделать?

- решить проблему с тайм-лайнами (их не было как сущности);
- засаппортить оба формата определения времени (**cron** и **every**);
- полностью переписать существующее решение;
- подготовиться к релизу полноценного гема (уже зарелизил, и **@sharombola** уже поймал первый баг);
- сделать доку (не успел, но сделал);

Sidekiq::Portal - Какую проблему решает?

- у нас есть **Sidekiq::Scheduler** (а еще **Sidekiq::Cron**)

Sidekiq::Portal - Какую проблему решает?

- у нас есть **Sidekiq::Scheduler** (а еще **Sidekiq::Cron**)
- мы обожаем писать тесты и делаем это хорошо (да?)

Sidekiq::Portal - Какую проблему решает?

- у нас есть **Sidekiq::Scheduler** (а еще **Sidekiq::Cron**)
- мы обожаем писать тесты и делаем это хорошо (да?)
- мы любим **Timecop**, особенно **Timecop.travel** и **Timecop.freeze**

Sidekiq::Portal - Какую проблему решает?

- у нас есть **Sidekiq::Scheduler** (а еще **Sidekiq::Cron**)
- мы обожаем писать тесты и делаем это хорошо (да?)
- мы любим **Timecop**, особенно **Timecop.travel** и **Timecop.freeze**
- но Timecop не любит scheduler'ы 😞 - джобки не запускаются

Sidekiq::Portal - Какую проблему решает?

- у нас есть **Sidekiq::Scheduler** (а еще **Sidekiq::Cron**)
- мы обожаем писать тесты и делаем это хорошо (да?)
- мы любим **Timecop**, особенно **Timecop.travel** и **Timecop.freeze**
- но Timecop не любит scheduler'ы 😞 - джобки не запускаются
- на рынке Github BAZAR нет решений



Так а в чем проблема?

- `BalanceWithdrawJob ({ every: '15m' })`
- `Timecop.travel(Time.curren + 2.hours)`

A

B

(ничего не делаем)

- `BalanceWithdrawJob ({ every: '15m' })`
- `Timecop.travel(Time.curren + 2.hours)`

A

B

(ничего не делаем)

A

B

(должно было случиться? - выполним разочек)

- `BalanceWithdrawJob ({ every: '15m' })`
- `Timecop.travel(Time.current + 2.hours)`

A

B

(ничего не делаем)

A

B

(должно было случиться? - выполним разочек)

A

B

(посчитаем, сколько раз должно было - столько раз и выполним)



Проблема?



Time . current

- `BalanceWithdrawJob ({ every: '15m' })`
- `Timecop.travel(Time.curren + 2.hours)`

A

B

1) перехватываем `Timecop.travel`

- `BalanceWithdrawJob ({ every: '15m' })`
- `Timecop.travel(Time.curren + 2.hours)`

A B

1) перехватываем `Timecop.travel`

A B



2) строим таймлайн согласно планировщику

- `BalanceWithdrawJob ({ every: '15m' })`
- `Timecop.travel(Time.curren + 2.hours)`

A B

1) перехватываем `Timecop.travel`

A B



2) строим таймлайн согласно планировщику

A B



3) прыгаем по таймпоинтам и выполняем все джобки в очереди

```
Timecop.freeze(time_point) { run_job }
```


Пример (ставим гемчик)

```
gem 'sidekiq', '>= 5' # runtime dependency
gem 'timecop', '~> 0.9' # runtime dependency

group :test do
  gem 'rspec'
  gem 'sidekiq-portal'
end
```

```
bundle install
```

```
require 'timecop' # runtime dependency
require 'sidekiq' # runtime dependency
require 'sidekiq/api'
require 'sidekiq/testing'

require 'sidekiq_portal'
```

Пример (наша джобка)

```
class HookExampleJob < ApplicationJob
  def perform
    GLOBAL_HOOK_INTERCEPTOR << Time.current # intercept current time
  end
end
```

```
:schedule:
  HookExample:
    every: '15m'
```

Пример (конфигурируем)

```
# portal configuration
Sidekiq::Portal.setup! do |config|
  config.default_timezone = 'UTC'
  config.retries_count = 1

  # pre-defined sidekiq-scheduler configs (Rails example)
  config.scheduler_cofnig = Rails.application.config_for(:sidekiq)[:schedule]

  # manual sidekiq-scheduler configs
  config.scheduler_config = {
    LoolJob: { every: '15m' },
    kek_job: { cron: '0 * * * *', class: :KekJob }
  }
end

# global state clearing logic
RSpec.configure do |config|
  config.before { Sidekiq::Worker.clear_all }
  config.after { Timecop.return }
  config.after { Sidekiq::Portal.reload! }
end
```

Пример (тестим)

```
RSpec.describe 'HookExampleJob sheduler plan' do
  specify 'scheduled?' do
    stub_const('GLOBAL_HOOK_INTERCEPTOR', [])
    expect(GLOBAL_HOOK_INTERCEPTOR.count).to eq(0) # => true

    # do some magic 🐱
    Timecop.travel(Time.current + 2.hours)

    expect(GLOBAL_HOOK_INTERCEPTOR.count).to eq(8) # => true (🐱 magic)

    puts GLOBAL_HOOK_INTERCEPTOR # 🐱
    # => outputs:
    # 2019-12-24 03:05:39 +0300 (+15m) (Time.current from HookExampleJob#perform)
    # 2019-12-24 03:20:39 +0300 (+15m)
    # 2019-12-24 03:35:39 +0300 (+15m)
    # 2019-12-24 03:50:39 +0300 (+15m)
    # 2019-12-24 04:05:39 +0300 (+15m)
    # 2019-12-24 04:20:39 +0300 (+15m)
    # 2019-12-24 04:35:39 +0300 (+15m)
    # 2019-12-24 04:50:39 +0300 (+15m)
  end
end
```

Sidekiq::Portal - Что дальше?

- новый режим тестирования `Sidekiq::Testing.portal!`
- добавить поддержку ретраев;
- написать специальные rspec-матчеры;
- добавить поддержку `Sidekiq::Worker` и `Sidekiq::Cron`;
- рандомизировать порядок выполнения джобок, которые должны запускаться в одно и тоже время;
- и всякие мелочи (в README.md есть ROADMAP);

Спасибо за внимание!

Sidekiq::Portal

https://github.com/0exp/sidekiq_portal