



Blazingfast™ deployment with mina

STJEPAN HADJIĆ, RAILS TEAM LEAD
STJEPAN.HADJIC@INFINUM.CO



20-PERSON RAILS TEAM



WHAT DOES IT MEAN TO DEPLOY A WEB APPLICATION?

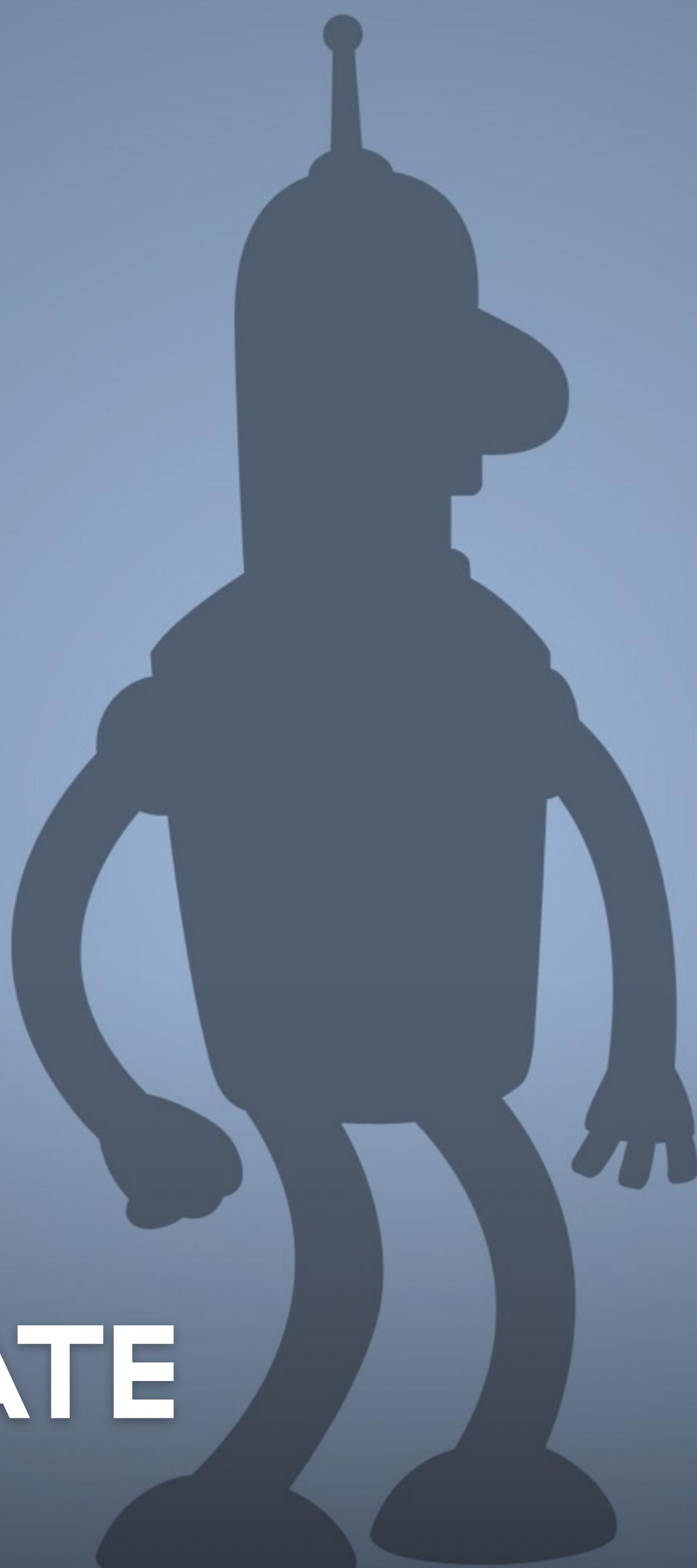
**New *version* of your
application**

SFTP / **SCP**

```
$ ssh deploy@server.com  
$ git pull  
$ rake db:migrate  
$ restart_application
```

production quick fixes

reverting changes?



AUTOMATE



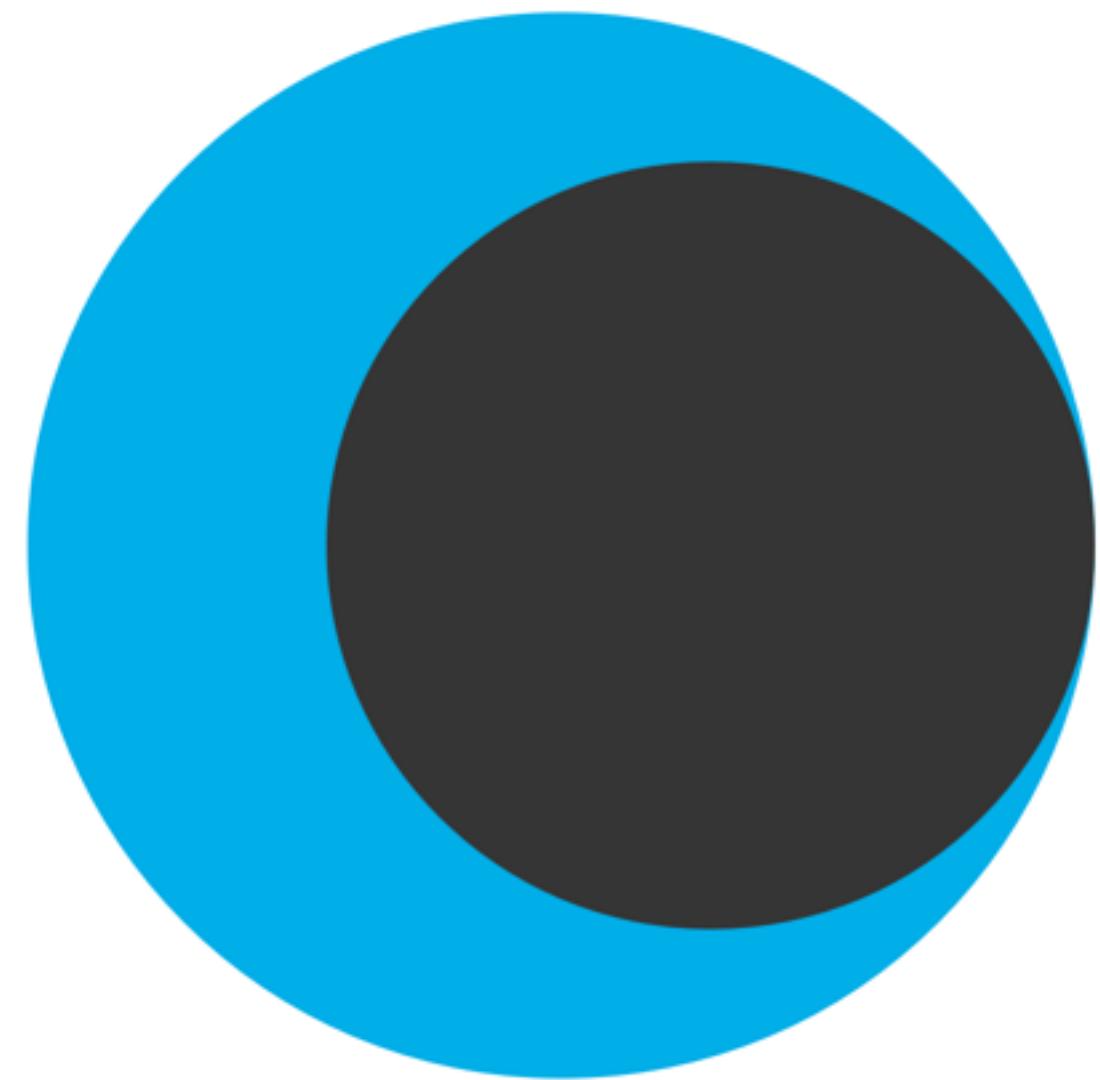


WHY MINA?



**Deploy several times a
day**

FIRST

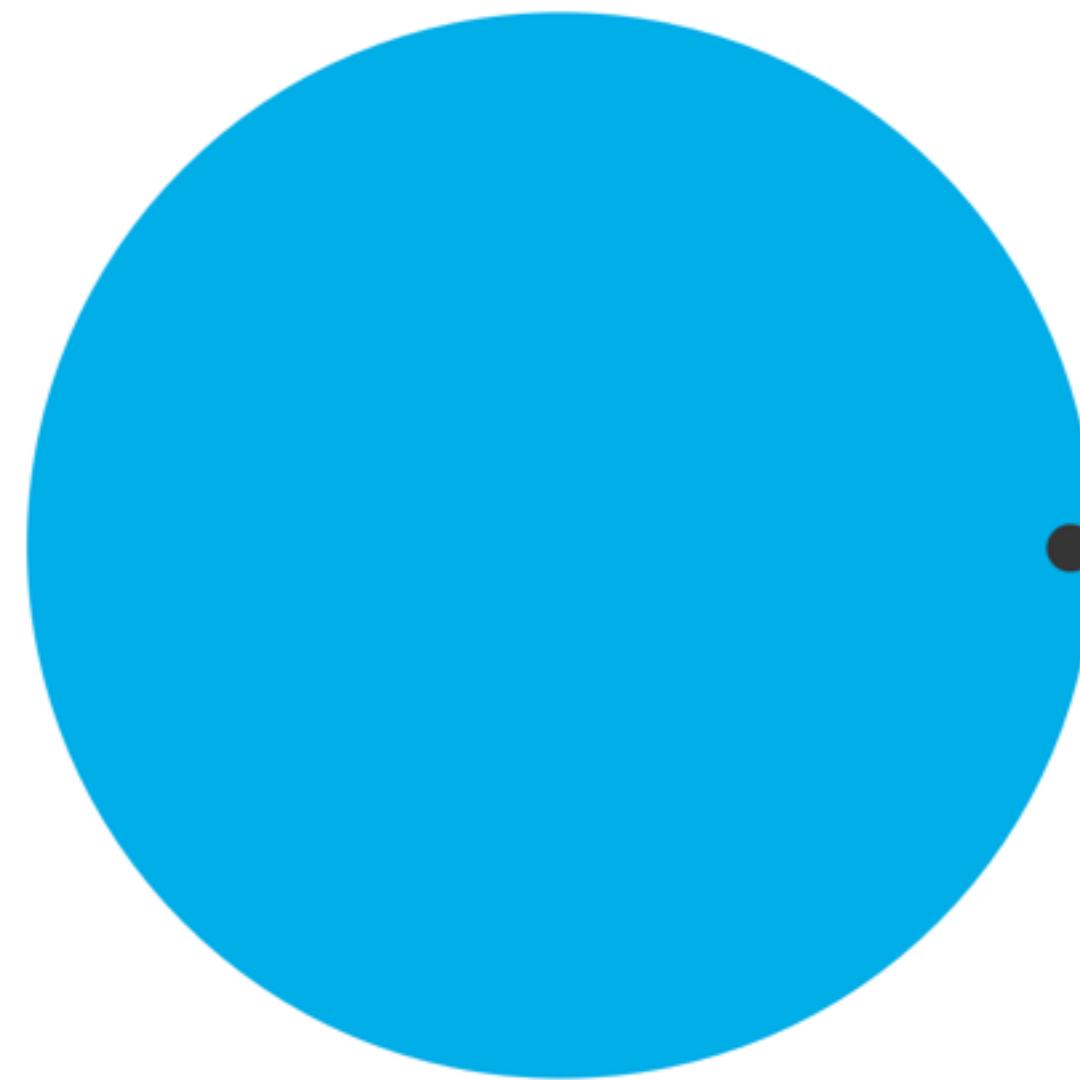


902
sec



665
sec

SUBSEQUENT

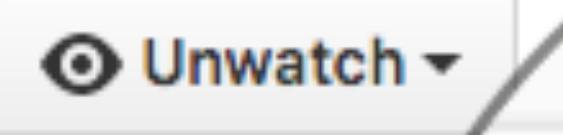


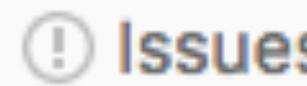
140
sec



5
sec

 [mina-deploy / mina](#)

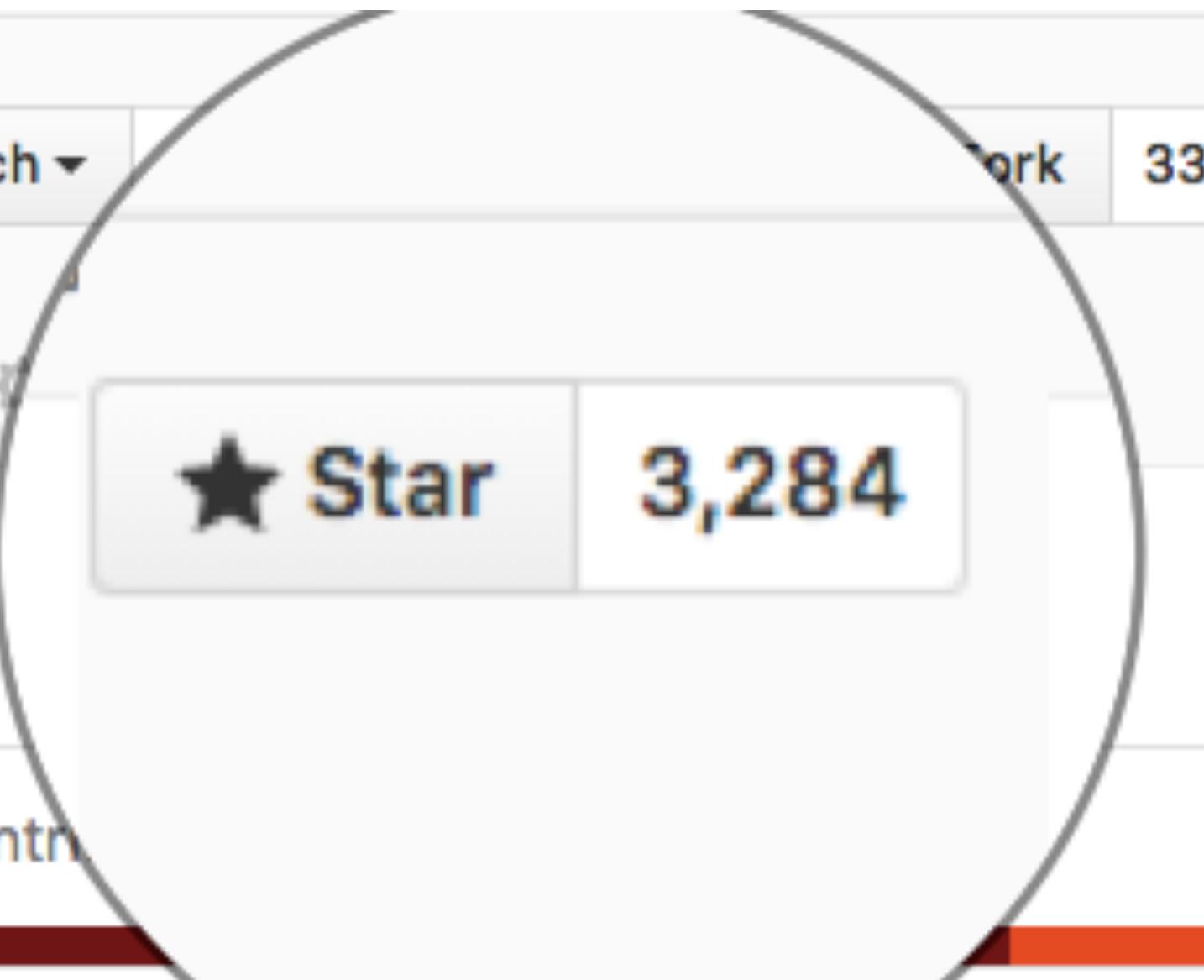
Code climate 3.7 

 Code  Issues 4  Pull requests 0  Projects 0  Pulse  Graphs

 Star 3,284

Blazing fast application deployment tool. <https://github.com/mina-deploy/mina> — Edit

 777 commits  1 branch  40 releases  78 contri





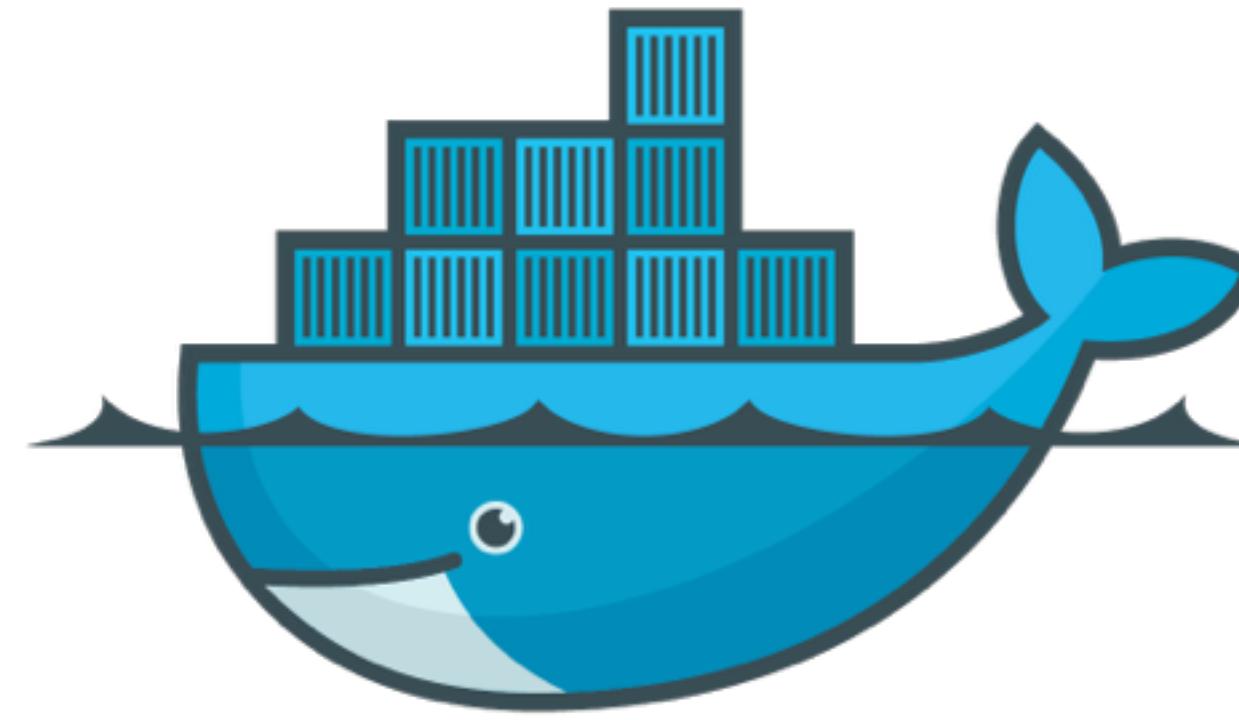


WHAT MINA IS NOT

**System
automation
platform**



Virtualization
tool



docker



**Your server must
already be set up!**

Key-based SSH

**Required software
already installed (git,
nginx etc.)**

**Deploy user set up
correctly**





APPLICATION DEPLOYMENT
A JOY

Wrapper around `rake`

Shell **script** generator

**Executes the script in
one SSH connection**



FILE STRUCTURE

```
➜ demo_deploy tree -dL 2
current ->../../demo_deploy/releases/13
├── releases
│   ├── 10
│   ├── 11
│   ├── 12
│   ├── 13
│   ├── 8
│   └── 9
└── scm
    ├── branches
    ├── hooks
    ├── info
    ├── objects
    └── refs
├── shared
│   ├── log
│   ├── public
│   ├── tmp
│   └── vendor
└── tmp
```



EASY TO USE

\$ mina init

----> Created ./config/deploy.rb
Edit this file, then run `mina setup` after.

```
set :repository, 'git@github.com:...'

set :domain, 'server.com'
set :user, 'deploy'

set :deploy_to, '/path/to/project/root'
set :rails_env, 'production'

task :deploy do
  deploy do
    ...
  end
end
```

\$ mina setup

```
--> Setting up ../demo_deploy
./demo_deploy
├── config
│   └── deploy.rb
├── releases
└── shared
    ├── log
    ├── public
    │   └── assets
    ├── tmp
    │   └── cache
    └── vendor
        └── bundle
```

10 directories, 1 file

Elapsed time: 0.47 seconds

\$ mina deploy

```
- - - - > Creating a temporary build path
- - - - > Fetching new git commits
- - - - > Using git branch 'master'
             Cloning into '.'...
             done.
- - - - > Using this git commit
             Stjepan Hadjic (52b31e5):
             > lots of things
- - - - > Symlinking shared paths
- - - - > Cleaning up old releases (keeping 5)
- - - - > Deploy finished
- - - - > Building
- - - - > Moving build to ~/demo_mina/releases/3
- - - - > Build finished
- - - - > Launching
- - - - > Updating the ~/demo_mina/current symlink
- - - - > Restarting application
- - - - > Done. Deployed version 3
```

Elapsed time: 3.152 seconds



RELIABLE

```
desc 'Deploys the current version to the server.'
task deploy: :environment do
  deploy do
    invoke :'git:clone'
    invoke :'deploy:link_shared_paths'
    invoke :'bundle:install'
    invoke :'rails:db_migrate'
    invoke :'rails:assets_precompile'
    invoke :'deploy:cleanup'

    on :launch do
      invoke :restart_application
    end
  end
end
```

```
(  
    checks &&  
    lock &&  
    create build dir &&  
    run default stage &&  
    move build dir to release dir &&  
    run build stage &&  
    link current to release dir &&  
    run launch stage &&  
    unlock  
) || clean up failed release
```



FAST

Elapsed time: **3 . 152** seconds



WHY REWRITE?



BAD STATE OF CODE

The background is a dark, abstract scene. It features several glowing blue energy fields or plasma-like structures that resemble galaxies or nebulae. Interspersed among these blue forms are several pairs of glowing red eyes, which appear to be looking in different directions. The overall atmosphere is mysterious and otherworldly.

UNFAMILIAR CODEBASE

Desire to make the
DSL **clearer**

```
set_default :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  queue %{
    echo "----> Loading rbenv"
    #{echo_cmd %{export RBENV_ROOT="#{rbenv_path}"}}
    #{echo_cmd %{export PATH="#{rbenv_path}/bin:$PATH"}}
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set_default :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  queue %{
    echo "----> Loading rbenv"
    #{echo_cmd %{export RBENV_ROOT="#{rbenv_path}"}}
    #{echo_cmd %{export PATH="#{rbenv_path}/bin:$PATH"}}
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  queue %{
    echo "----> Loading rbenv"
    #{echo_cmd %{export RBENV_ROOT="#{rbenv_path}"}}
    #{echo_cmd %{export PATH="#{rbenv_path}/bin:$PATH"}}
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  queue %{
    echo "----> Loading rbenv"
    #{echo_cmd %{export RBENV_ROOT="#{rbenv_path}"}}
    #{echo_cmd %{export PATH="#{rbenv_path}/bin:$PATH"}}
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  comment %{Loading rbenv}
  queue %{
    #{echo_cmd %{export RBENV_ROOT="#{rbenv_path}"}}
    #{echo_cmd %{export PATH="#{rbenv_path}/bin:$PATH"}}
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  comment %{Loading rbenv}
  queue %{
    #{echo_cmd %{export RBENV_ROOT="#{rbenv_path}"}}
    #{echo_cmd %{export PATH="#{rbenv_path}/bin:$PATH"}}
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  comment %{Loading rbenv}
  command %{export RBENV_ROOT="#{rbenv_path}"}
  command %{export PATH="#{rbenv_path}/bin:$PATH"}
  command %{
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  comment %{Loading rbenv}
  command %{export RBENV_ROOT="#{rbenv_path}"}
  command %{export PATH="#{rbenv_path}/bin:$PATH"}
  command %{
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  comment %{Loading rbenv}
  command %{export RBENV_ROOT="#{fetch(:rbenv_path)}"}
  command %{export PATH="#{fetch(:rbenv_path)}/bin:$PATH"}
  command %{
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  comment %{Loading rbenv}
  command %{export RBENV_ROOT="#{fetch(:rbenv_path)}"}
  command %{export PATH="#{fetch(:rbenv_path)}/bin:$PATH"}
  command %{
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  queue! %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  comment %{Loading rbenv}
  command %{export RBENV_ROOT="#{fetch(:rbenv_path)}"}
  command %{export PATH="#{fetch(:rbenv_path)}/bin:$PATH"}
  command %{
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  command %{eval "$(rbenv init -)"}
end
```

```
set :rbenv_path, "$HOME/.rbenv"

task :'rbenv:load' do
  comment %{Loading rbenv}
  command %{export RBENV_ROOT="#{fetch(:rbenv_path)}"}
  command %{export PATH="#{fetch(:rbenv_path)}/bin:$PATH"}
  command %{
    if ! which rbenv >/dev/null; then
      echo "! rbenv not found"
      exit 1
    fi
  }
  command %{eval "$(rbenv init -)"}
end
```



MORE POWERFUL

```
task :deploy do
  run :local do
    command %{ask_for_permission}
  end

  deploy do
    ...
  end

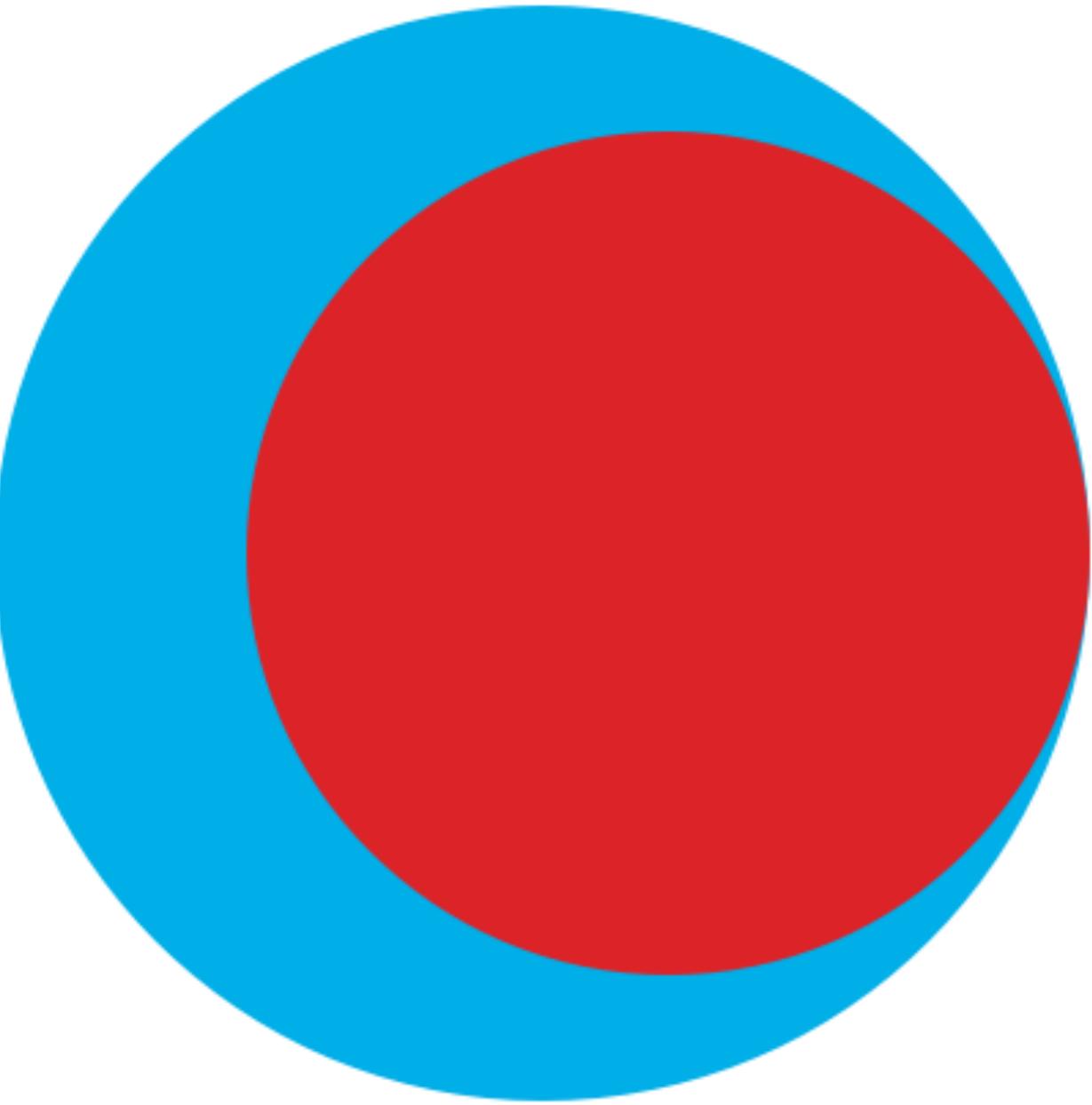
  run :local do
    command %{say 'Jobs done!'}
  end

  run :backend do
    command %{notify_elk}
  end
end
```

```
task :deploy do
  invoke :'git:ensure_pushed'
end
```

Faster for rails
applications

FIRST

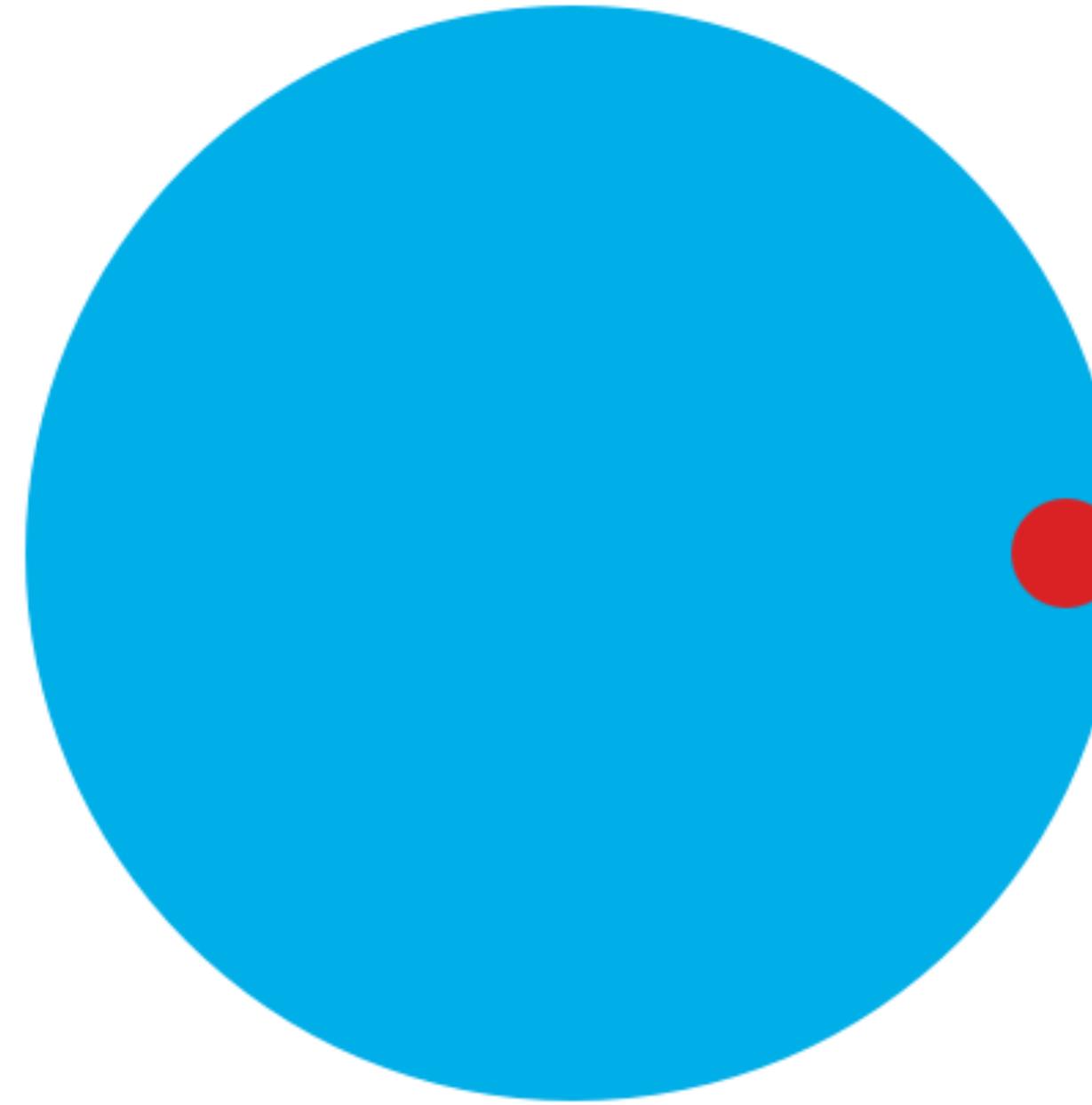


143
sec



111.6
sec

SUBSEQUENT



35.6
sec



3.45
sec

integrated plugins

3rd party plugins



DRAWBACKS

Parallel executions

Server roles

Only what your **deploy
user can do**

when to use mina?



CONTINUOS DEPLOYMENT

```
$ gem install mina --no-document  
$ mina staging ssh_keyscan_domain  
$ mina staging deploy
```

Server **staging**[Delete server](#)**Server name**

staging

Deploy commands**1** `gem install mina -v '1.0.0.rc2' --no-document`**2** `gem install mina-data_sync -v '1.0.0.rc1' --no-document`**3** `mina staging ssh_keyscan_domain`**4** `mina staging deploy`[Change deploy commands](#)



INFINUM

We're an independent
design & development
agency.



infinum.co/our-stuff



#AndroidSweets



#iOSCocoaTreats



#RubyDrops



#Design Slices



#FrontendCookies



#RubyDrops



Questions?

Visit infinum.co or find us on social networks:





Hit me up on @_Beast_

Visit infinum.co or find us on social networks:

