

02

OPEN ORIENTED

凹凸实验室

持续集成Travis CI

持续集成(Continuous Integration), 简称CI, 意思是, 在一个项目中, 任何人对代码库的任何改动, 都会触发CI服务器自动对项目进行构建, 自动运行测试, 甚至自动部署到测试环境。这样做的好处就是, 随时发现问题, 随时修复。因为修复问题的成本随着时间的推移而增长, 越早发现, 修复成本越低。

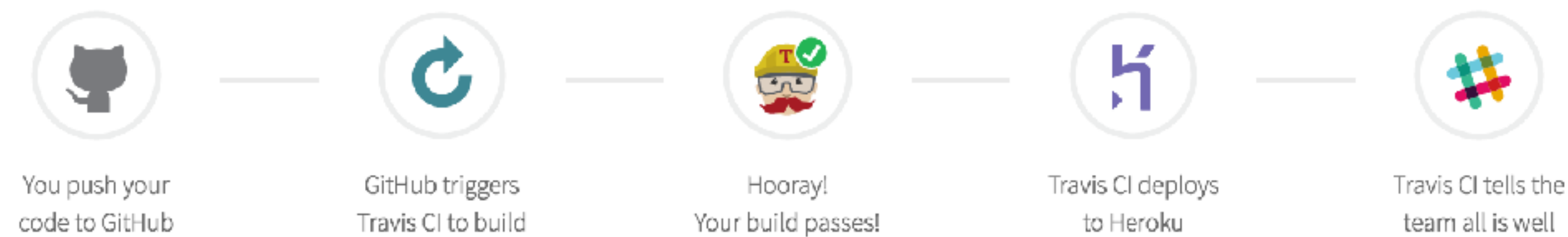


Gitlab CI

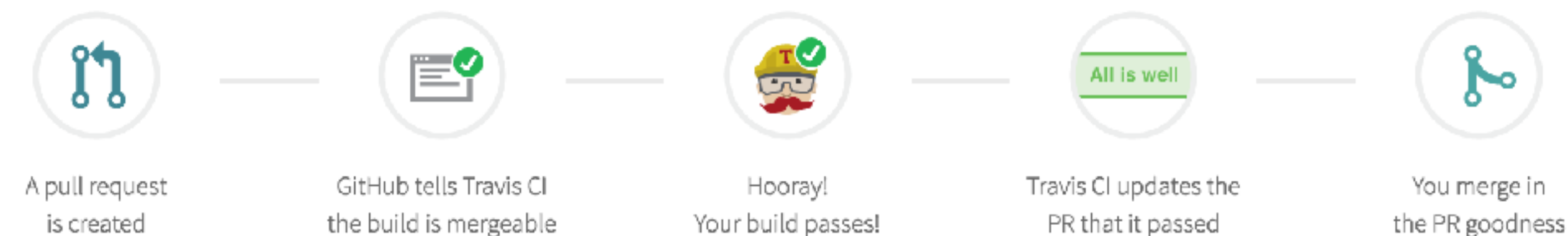
持续集成Travis CI

Travis CI是在线托管的CI服务，集成了GitHub并且能执行自动化的构建和测试操作，它对公有的GitHub仓库免费。

Branch build flow



Pull request build flow



Travis CI 语言支持度

Language-specific Guides

ANDROID

C

C#

C++

CLOJURE

CRYSTAL

D

DART

ERLANG

ELIXIR

F#

GO

GROOVY

HASKELL

HAXE

JAVA

JAVASCRIPT (WITH NODE.JS)

JULIA

OBJECTIVE-C

PERL

PERL6

PHP

PYTHON

R

RUBY

RUST

SCALA

SMALLTALK

VISUAL BASIC

1. 用你的GitHub账号登录Travis-CI， 确认接受访问GitHub的权限。
2. 登录之后， Travis-CI就会同步你GitHub账号的仓库， 之后打开个人页面并给你想要构建的项目启用Travis-CI。
3. 添加 `.travis.yml` 文件到你项目根目录下， Travis-CI会按照`.travis.yml`里的内容进行构建。
4. 把 `.travis.yml` push到你的GitHub上以触发Travis-CI进行构建。
5. 构建状态页面查看项目是否构建成功。

.travis.yml文件

```
language: node_js # 声明构建语言环境

notifications:      # 每次构建的时候是否通知，如果不想收到通知邮箱，就设置false
  email: false

#sudo: false # 开启基于容器的Travis CI任务，让编译效率更高

node_js: # node测试版本，可以多个
# - "7"
- "6"
# - "5"

cache: # 构建缓存文件，方便开始构建测试
  directories:
    - node_modules

branches: #分支
  only:
    - travis-ci

services: #构建测试所需服务
- docker

before_install: #install前触发
install: #环境准备，如生成可执行文件、配置环境变量

before_script: #执行测试脚本前触发
after_success: #脚本执行成功后触发
after_failure: #脚本执行失败后触发
after_script: #脚本执行后触发

script:
- node index.js
- sh .travis/test.sh

env: #集成测试所需环境
- DB=postgres
- SH=bash
- PACKAGE_VERSION="1.0.*"
- mongodb

deploy: # 部署
```

- [Travis-CI官方文档](#)
- [如何简单入门使用Travis-CI持续集成](#)
- [travis-ci-examples](#)
- [GitLab-CI文档](#)

T H A N K S
FOR YOUR WATCHING

