

# CI-CD Tools Comparison

CIRCLECI VS TRAVIS-CI VS JENKINS

# 1. CircleCI - Introduction

- A cloud based system: No dedicated server required no administration needed
- CircleCI caches requirements installation. It checks 3rd party dependencies instead of constant installations of the environments needed
- Requires minimal configuration and adjustments for its execution. Uses a YAML file for configuration.
- It is compatible with the following software's:
  - Python, Node.js, Ruby, Java, Go, etc.
  - Ubuntu (12.04, 14.04), Mac OS X (paid accounts)
  - Github, Bitbucket
  - AWS, Azure, Heroku, Docker, dedicated server
  - Jira, HipChat, Slack

# Pro's and Con's

## 1. Pros:

- Starts as soon as the code is pushed to the repository(Fast start)
- Lightweight, easily readable YAML config
- You do not need any dedicated server to run CircleCI

## 2. Cons:

- Supports only 2 versions of Ubuntu for free (12.04 & 14.04) and MacOS as a paid part
- Also, while being a cloud-based system is a plus from one side, it can also stop supporting any software, and you won't be able to prevent that

## 2. Travis CI - Introduction

- Travis CI and CircleCI are almost the same
- Like CircleCI, Travis CI also is cloud based, uses a YAML file for configuration and both support docker to execute tests.

### **So what does Travis offer that CircleCI doesn't?**

- Option to run tests on Linux and Mac OS X at same time
- Supports more languages out of the box:
- Support of build matrix

# Pro's and Con's

## 1. Pros:

- Build matrix out of the box
- Fast start
- Lightweight YAML config
- Free plan for open-sourced projects
- No dedicated server required

## 2. Cons:

- Price is higher compared to CircleCI, no free enterprise plan
- Customization (for some stuff you'll need 3rd parties)

# Pro's and Con's

## 1. Pros:

- Price : Its free
- Flexible customization
- Myriad of plugins
- Full control of the system

## 2. Cons:

- Dedicated server (or several servers) are required. That results in additional expenses. For the server itself, DevOps, etc...
- Time needed for configuration / customization

### 3. Jenkins - Introduction

- Jenkins is a self-contained Java-based program, ready to run out-of-the-box, with packages for Windows, Mac OS X and other Unix-like operating systems
- With hundreds of plugins in the Update Center, Jenkins integrates with practically every tool in the continuous integration and continuous delivery toolchain
- It has a wide range of job modes viz. Freestyle project, Pipeline, External Job, Multi-configuration project, Folder, GitHub Organization, Multibranch Pipeline
- Allows you to launch builds with various conditions.

# In Conclusion

- **What CI system to chose?** Well, that depends on your needs and the way you are planning to use it.
- **CircleCI** is recommended for small projects, where the main goal is to start the integration as fast as possible.
- **Travis CI** is recommended for cases when you are working on the open-source projects, that should be tested in different environments.
- **Jenkins** is recommended for the big projects, where you need a lot of customizations that can be done by usage of various plugins. You may change almost everything here, still this process may take a while. If you are planning the quickest start with the CI system Jenkins might not be your choice.