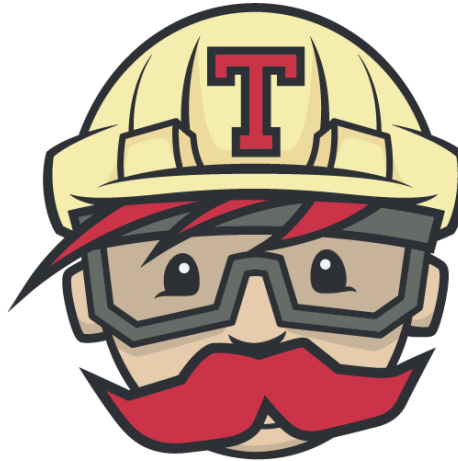




Northeastern

CircleCI vs Travis CI vs Jenkins



TEAM MEMBERS

- Ashish Aggarwal
- Sonali Chaudhari
- Renu Hadke
- Vikas Singh

Continuous Integration

- Continuous Integration (CI) is a software development practice that is based on a frequent integration of the code into a shared repository.
- Each check-in is then verified by an automated build.
- The main goal of continuous integration is to identify the problems that may occur during the development process earlier and more easily.
- If you integrate regularly — there is much less to check while looking for errors.
- That results in less time spent for debugging and more time for adding features.

Circle CI

CircleCI Pros:

- Fast start
- CircleCI has a free plan for enterprise projects
- It's easy and fast to start
- Lightweight, easily readable YAML config
- You do not need any dedicated server to run CircleCI

CircleCI Cons:

- CircleCI supports only 2 versions of Ubuntu for free (12.04 и 14.04) and MacOS as a paid part
- Despite the fact CircleCI do work with and run on all languages tt supports only the following programming languages "out of the box":Go (Golang), Haskell, Java, PHP, Python, Ruby/Rails, Scala
- Some problems may appear in case you would like to make customizations: you may need some 3rd party software to make those adjustments
- 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

Travis CI

Pros:

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

Cons:

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

Jenkins

Pros:

- Price (it's free)
- Customization
- Plugins system
- Full control of the system

Cons:

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

Jenkins vs Travis CI vs CircleCI

Jenkins:

- Powerful CI/CD tool for big projects
- It needs dedicated server with Jenkins installed
- Lots of plugins available for Jenkins to make it easier to use and manageable
- It also supports different job models like Freestyle, Pipeline, etc.,
- More customization option available with Jenkins

Travis CI:

- Cloud based CI/CD tool. No dedicated server is required to maintain this tool
- Suggested to use for open source projects
- Free for open source projects and paid subscription available for Enterprises
- We can build our own matrix (More customization than CircleCI)
- YAML configuration

CircleCI:

- Same like Travis CI – cloud based CI/CD tool
- Minimal configuration/adjustments are allowed
- Quick starting
- Free plans available for Individual and Enterprises
- Best fit for small projects to start the integration process very fast

Comparison

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.

Both of them:

- Have YAML file as a config
- Are cloud-based
- Have support of Docker to run tests

What does TravisCI 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: 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
- Support of build matrix

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.

Comparison

	Travis CI	CircleCI	Jenkins
Compatibility	Linux, Mac OS X, etc	Ubuntu, Mac OS X, etc	Windows, Mac OS X, Unix, etc
Supported Language	Python, Node.js, Ruby, Java, Go, etc	Crystal, Node.js, Go, Haskell, etc	Python, Ruby, Java, Android, C/C++, etc
SCM Tools support	Github	Github, Bitbucket	Github
Usage Plans	Free for open source projects, Paid for Enterprise	Free	Free
Server Machine	Cloud based	Cloud based	Server based
Customization Options	Less	Very less	More
Configuration	YAML	YAML	Fully customizable
Control on system	Very less	Very less	Full

Thank you!