

Assignment 6

TravisCi vs CircleCI vs Jenkins

Nandeep nelagondanahalli
Nithin Kartha
Vaishak P Dinesh

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.
 - There is also an option to set up inspection of the code style, cyclomatic complexity (low complexity makes the testing process more simple) and other checks.
- This helps to minimize the efforts of the person responsible for the code review, saves time, and improves the quality of the code

TravisCI

- **Travis CI** is a hosted, distributed continuous integration service used to build and test software projects hosted at GitHub.
- Open source projects may be tested at no charge via travis-ci.org.
- Private projects may be tested at travis-ci.com on a fee basis.
- TravisPro provides custom deployments of a proprietary version on the customer's own hardware.
- Written in Ruby.
- Travis CI is configured by adding a file named `.travis.yml`, which is a YAML format text file, to the root directory of the repository.
 - This file specifies the programming language used, the desired building and testing environment (including dependencies which must be installed before the software can be built and tested), hardware.

CircleCI

- CircleCI is a cloud-based system—no dedicated server required, and you do not need to administrate it. However, it also offers an on-prem solution as well.
- It has a free plan even for a business account
- Rest API—you have an access to projects, build and artifacts. The result of the build is going to be an artifact or the group of artifacts. Artifacts could be a compiled application or executable files (e.g. android APK) or metadata (e.g. information about the tests' success)
- CircleCI caches requirements installation. It checks 3rd party dependencies instead of constant installations of the environments needed
- You can trigger SSH mode to access container and make your own investigation (in case of any problems appear)
- CircleCI is compatible with:
 - Python, Node.js, Ruby, Java, Go, etc
 - AWS, Azure, Heroku, Docker, dedicated server
 - Github, Bitbucket, Jira, HipChat, Slack

Jenkins

- 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
- Various job modes: Freestyle project, Pipeline, External Job, Multi-configuration project, Folder, GitHub Organization, Multibranch Pipeline
- **Jenkins Pipeline:** A suite of plugins which supports implementing and integrating continuous delivery pipelines into Jenkins.
 - Pipeline provides an extensible set of tools for modeling simple-to-complex delivery pipelines “as code” via the Pipeline DSL
 - Allows you to launch builds with various conditions.
 - You can run Jenkins with Libvirt, Kubernetes, Docker, and others.

TravisCI and CircleCI

- TravisCI and CircleCI are pretty much the same out of the box

Similarities

- Have YAML file as a config file
- Are cloud-based
- Have support of Docker to run tests
- Similar setup and configuration steps

Differences

- TravisCI has the Option to run tests on Linux and Mac OS X at same time
- TravisCI 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
- TravisCI Supports build matrix
- Travis is slightly more expensive than CircleCI

TravisCI/CircleCI vs Jenkins: Similarities

- Supports a wide variety of languages out of the box
- Compatible with mac and other unix flavours
- Natively supports Github as a SCM

TravisCI vs CircleCI vs Jenkins: Differences

TravisCI/CircleCI	Jenkins
Cloud based	Needs a dedicated server
Offers less customizing options	Gives you a wide range of customization
YAML based configuration	Gives you a fully customizable configuration options
Less control on the system	You get full control on the system
	Offer different job models

Other CI/CD tools

- Codeship
- Teamcity
- GitLab CI
- Bamboo



Thank you !