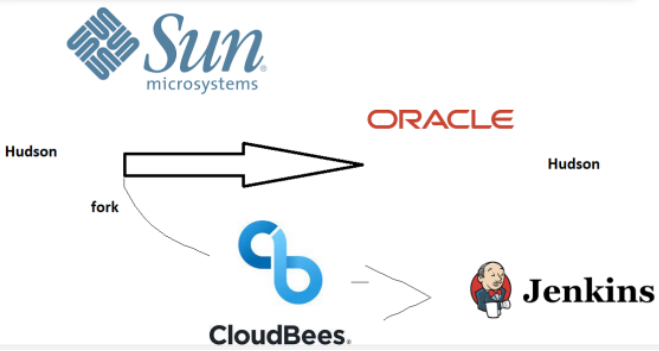
MENU ☰

FEBRUARY 25, 2023

DevOps Classroomnotes 25/Feb/2023

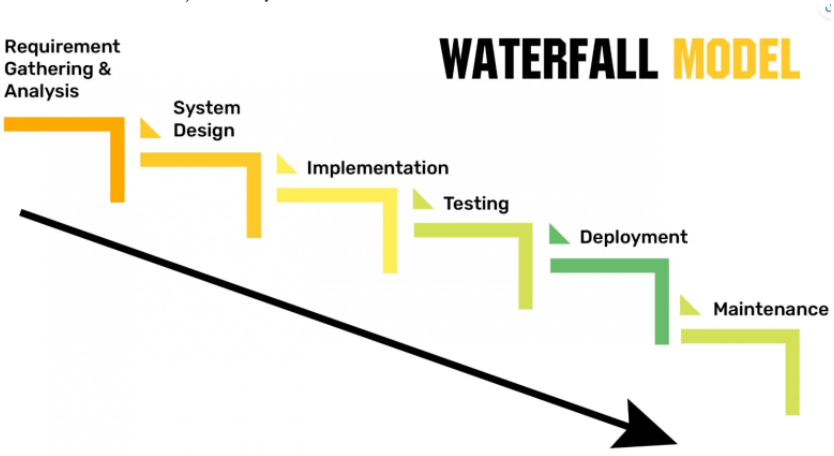
Jenkins

Story of Jenkins

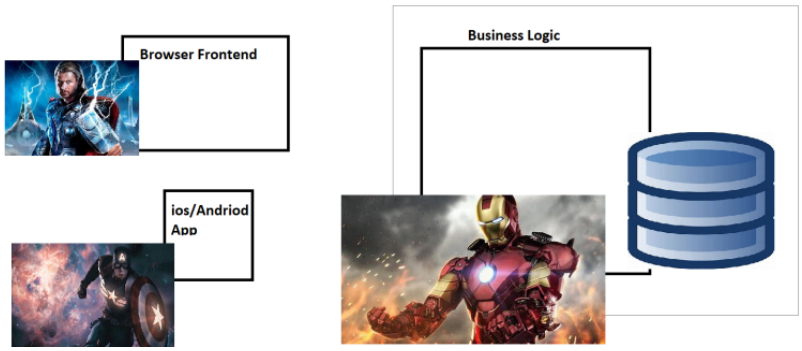


Need for Continuous Integration

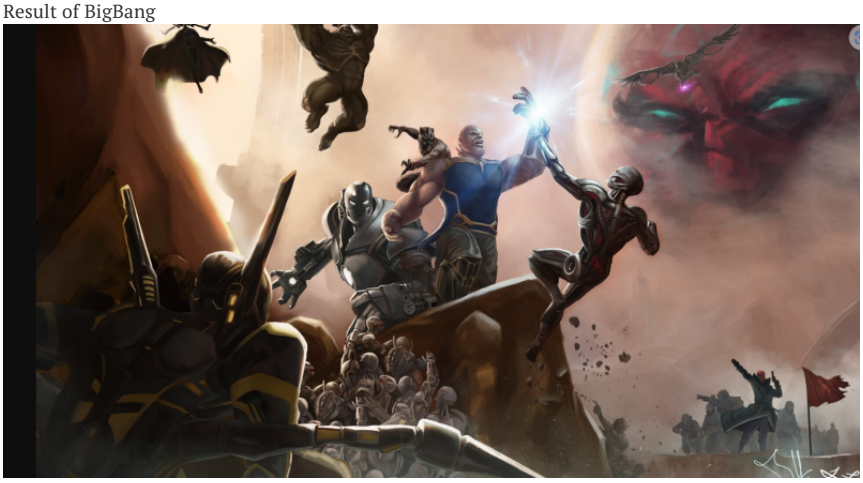
- Traditional Software Project Lifecycle



- Big Bang Integrations, were used by teams traditionally to combine components as each component was built by teams in isolation.



- Result of BigBang



- Continuous Integration (CI): In this we integrate all the components of the application from day 1. Ideally whenver developer pushes the code to remote repository we need to perform integration.
- To check whether CI is successful or not a small set of automated tests were developed i.e.
    - We build all the application components
    - integrate them and run this small set of tests which checks basic functionality of the system (Smoke Tests)

- Developers have started writing unit tests (test code which tests the application code)
- To measure the accuracy of unit tests generally we need a tool which measures coverage i.e. code coverage
- Code Coverage: Generally these are measure by tools
    - Line Coverage
    - branch coverage
    - symbol Coverage

- To verify the quality of code and whether it suits your organization's coding standards, Code Reviews are conducted.
- For coding standards and best practices we can use tools. The failures represented by these tools are generally referred as
    - Technical debt
    - Code Smell

- The tool for running tests is called as test harness tool or popularly know as **test runners**. For running unit tests
    - junit (java)
    - pytest (python)
    - jasmine (nodejs)
    - nunit (c-sharp)

- To check code coverage and code qualilty we have a tool which are generally referred as **Static Code Analysis Tools**. We will be using **Sonar Qube**
- We need a tool which whenever a code is pushed to vcs, builds the code, runs the unit tests and also generates Static Code Analysis report. These tools are referred as CI/CD Engines. Some of the popular CI/CD Engines are
    - Jenkins
    - Azure DevOps Pipelines
    - Travis CI
    - Team City
    - Bamboo

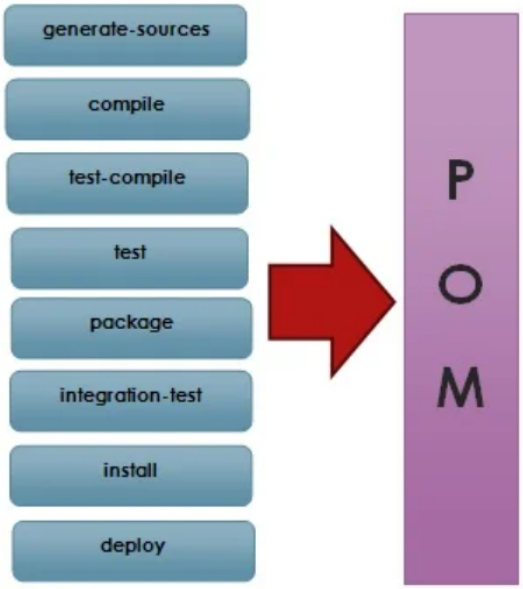
Build and Artifact

- Build represents a format/package into which your application code is transformed to run on enduser systems.
- To Generate Package we need to perform certain steps
    - compile
    - package

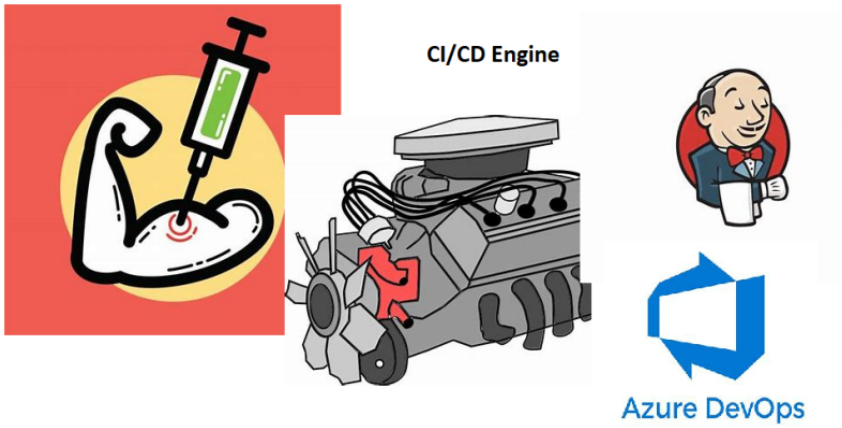
Building Java

- To make it simple to build java code , Apache foundation has created a tool called as **Ant**
- In Ant we configure the build using build.xml Refer Here
- Then a tool called as Maven was released which can perform
    - build
    - packaging
    - dependency management

    - documenation
- Maven goes with `Convention over Configuration` and all of them are around a file which is referred as `pom.xml` (project object model)
- Maven has a lifecycle
    - validate
    - compile
    - test
    - package
    - install
    - deploy
    - clean



- Every CI/CD Engine is Cron on steriods



Dependencies

- All the development projects depend on external libraries
    - Java: maven packages
    - node js: npm packages
    - python: pip packages
    - dotnet: nuget packages
- There is a need for managing dependencies

Leave a Reply

Enter your comment here...

About continuous learner

devops & cloud enthusiastic learner

VIEW ALL POSTS

PREVIOUS POST

DevOps Classroomnotes 25/Feb/2023

NEXT POST

DevOps Classroomnotes 26/Feb/2023