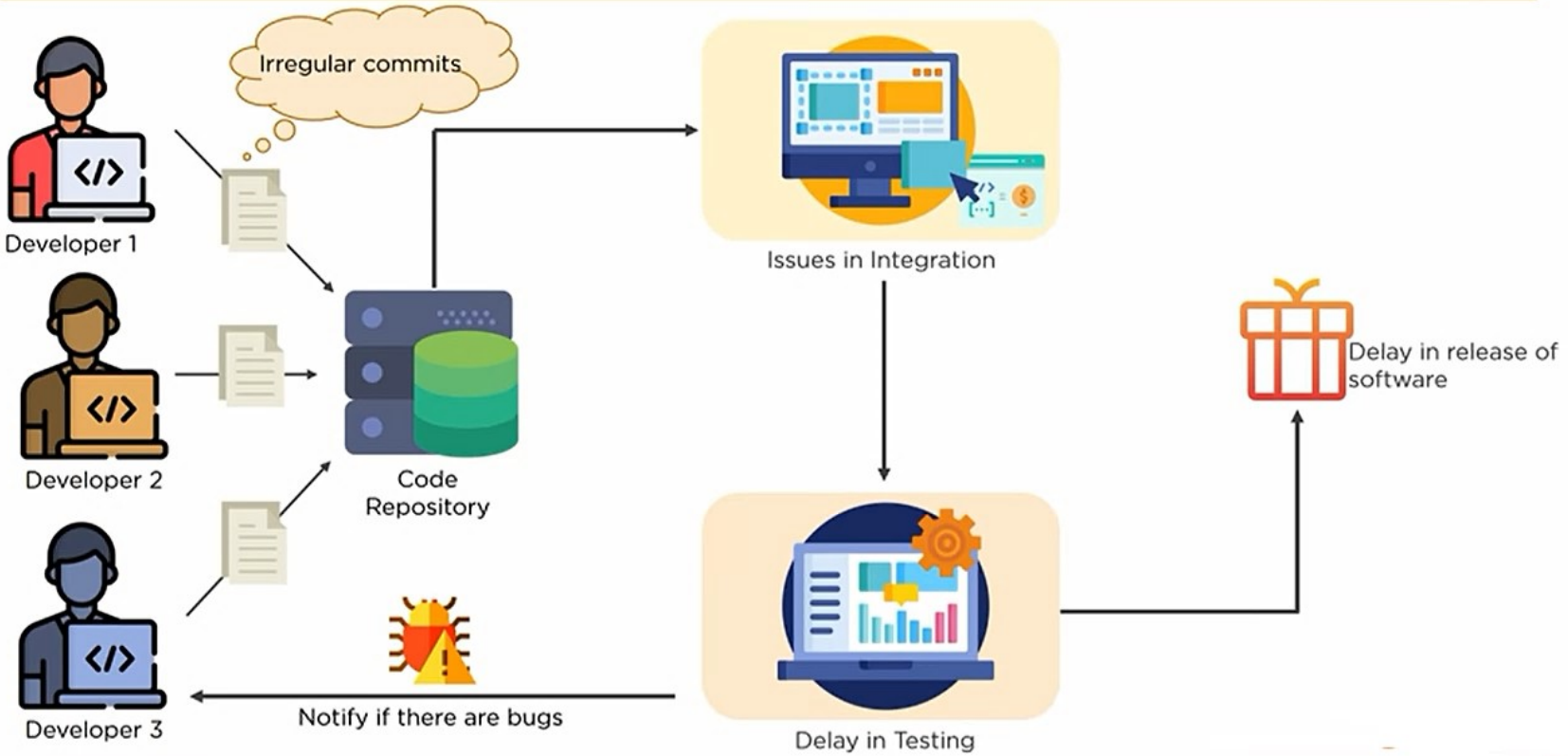
- Before Jenkins
- Issues before Jenkins
- What is Jenkins?
- What is Continuous Integration?

- Continuous Integration Tools
- Features of Jenkins
- Jenkins Architecture
- Jenkins Case Study


Before Jenkins

Before Jenkins



Before Jenkins

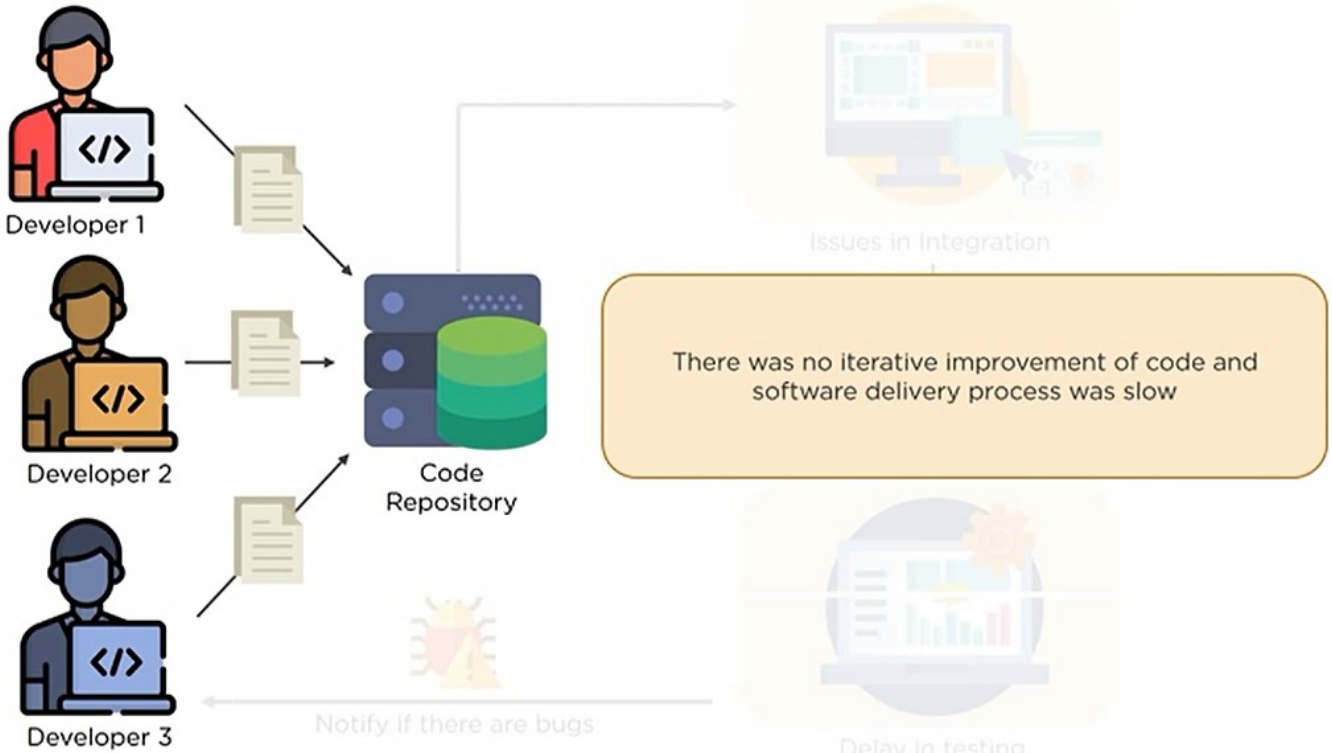
Developer 1

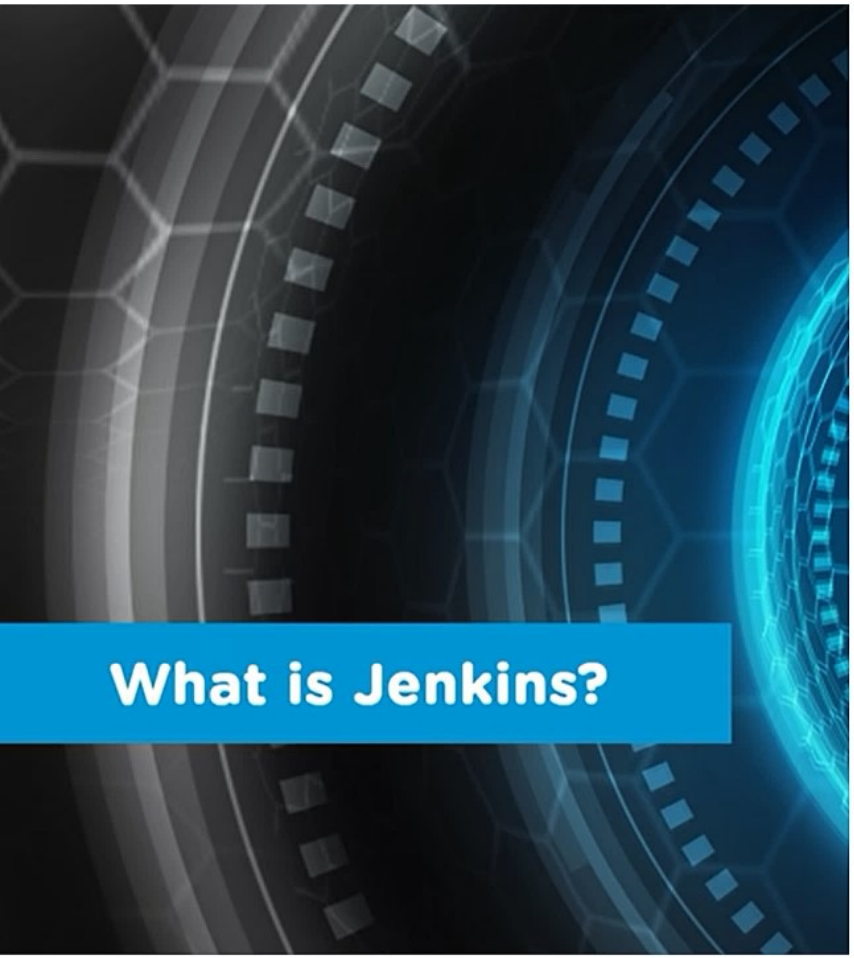
Developer 2

Developer 3

Developers had to wait till the entire software code was built and tested to check for errors

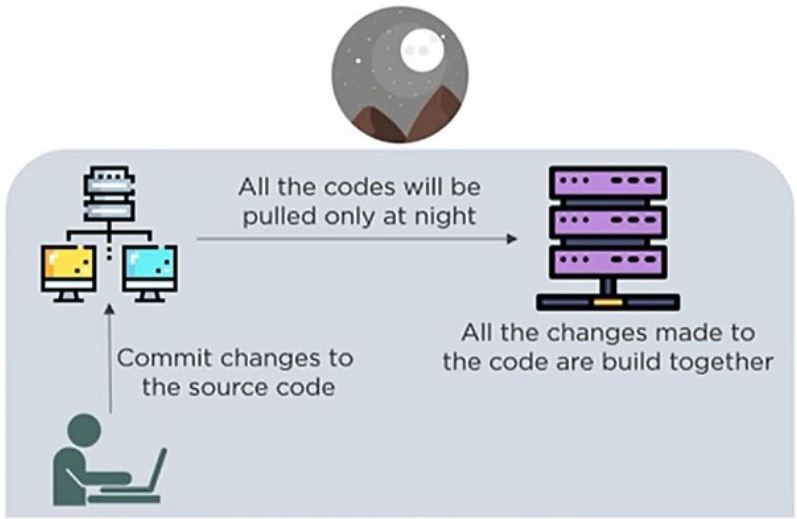
Before Jenkins



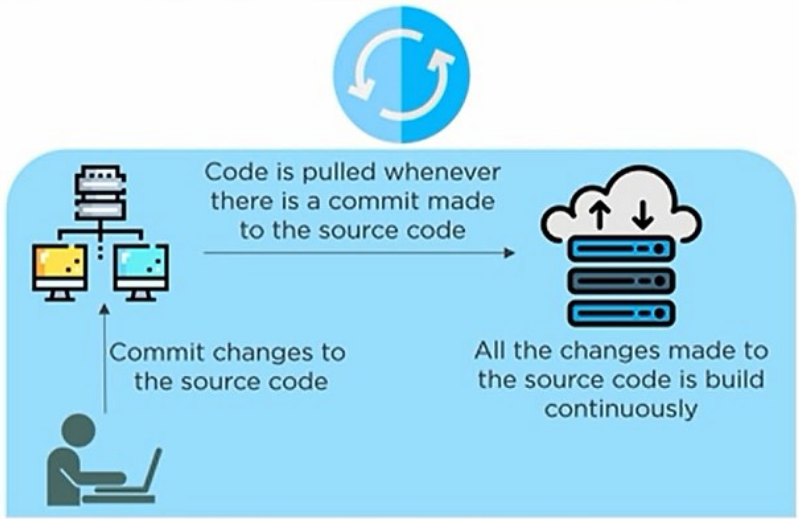

What is Jenkins?

What is Jenkins?

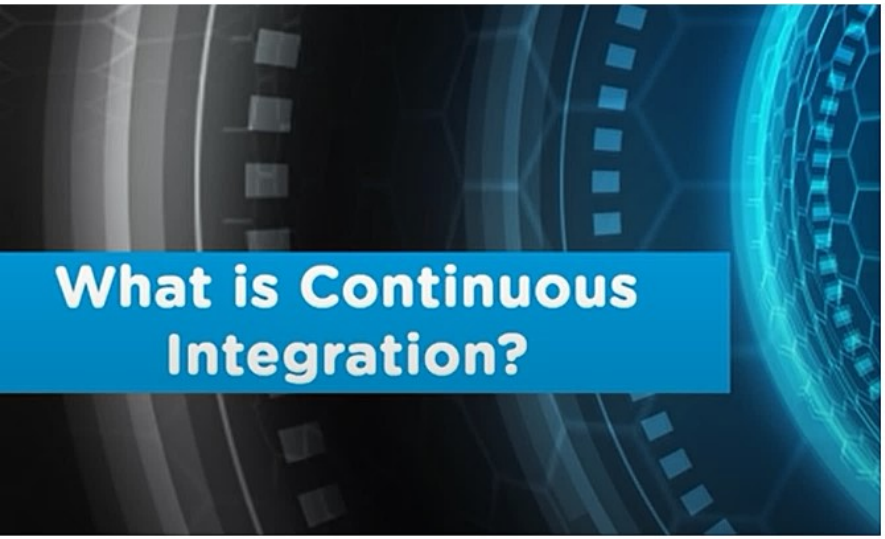
Jenkins is a Continuous Integration tool that allows continuous development, test and deployment of newly created codes



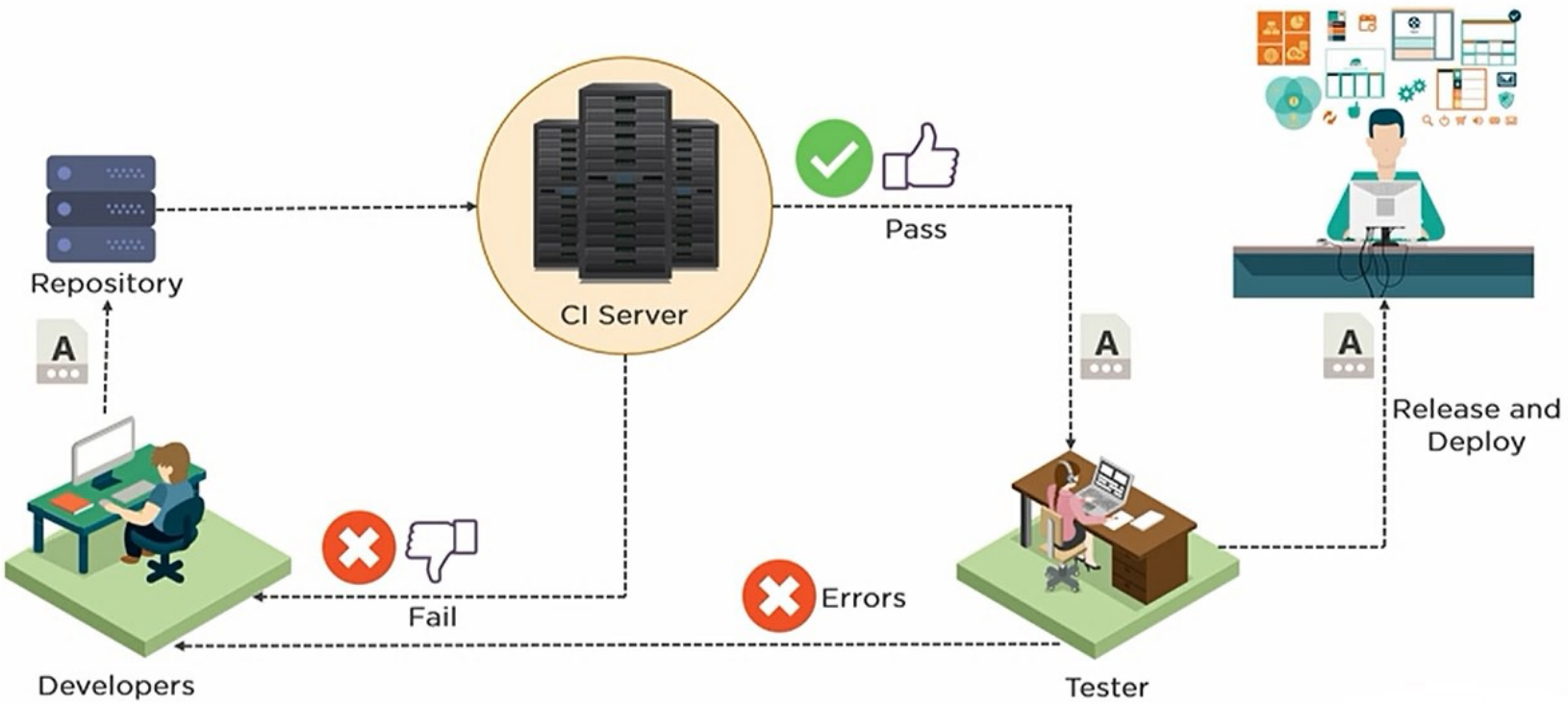
Nightly build and integration



Continuous build and Integration

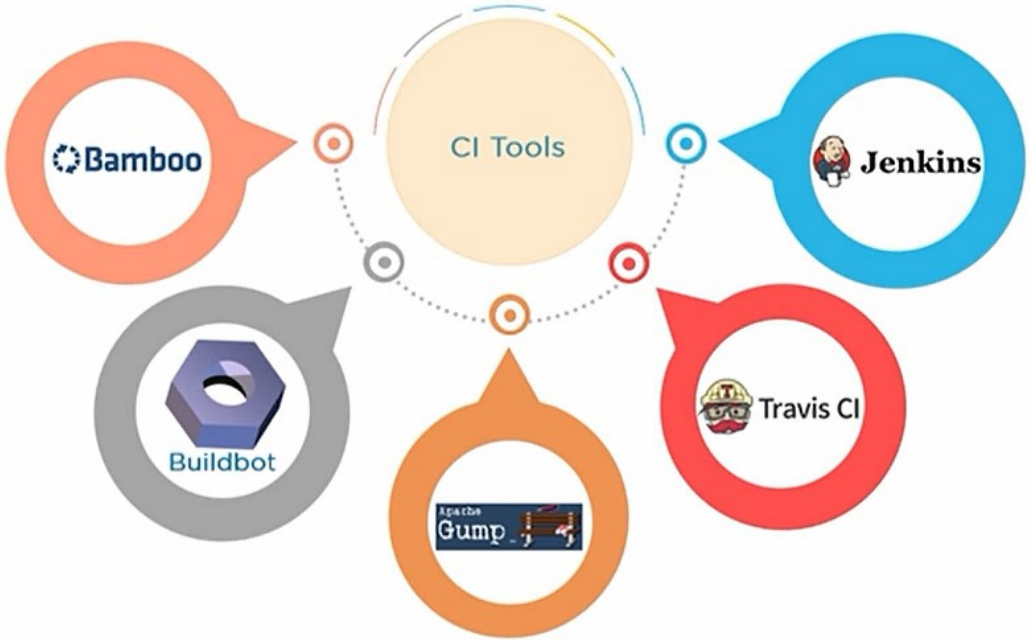

What is Continuous Integration?

What is Continuous Integration?



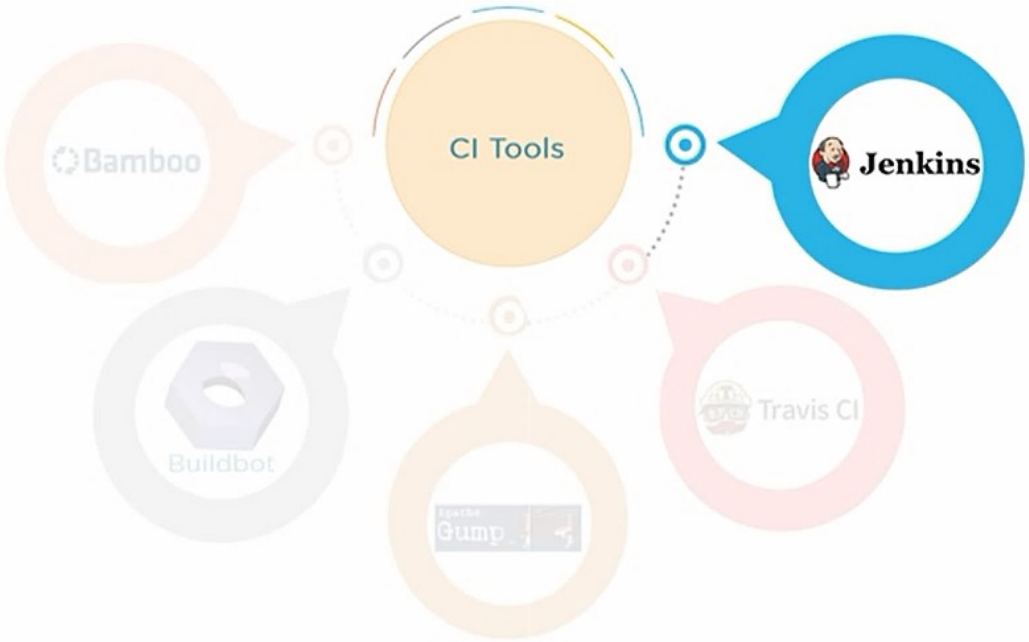

Continuous Integration
Tools

Continuous Integration Tools



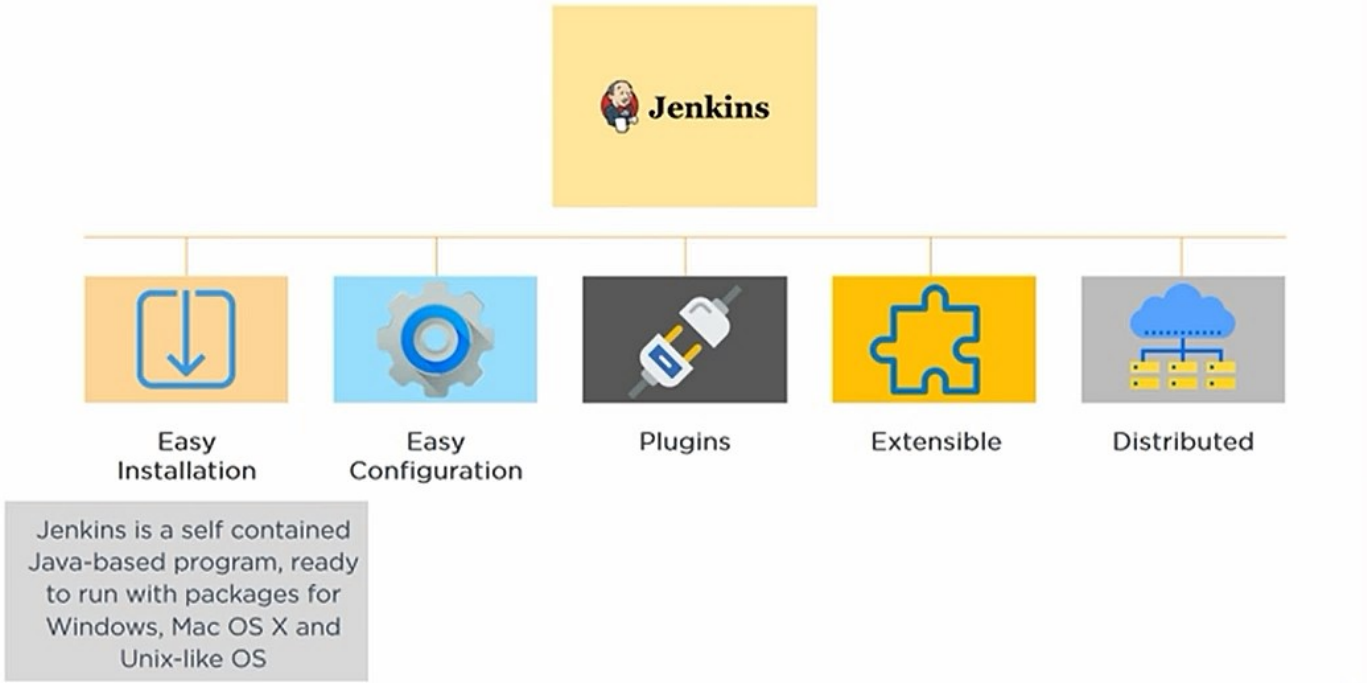
Continuous Integration Tools

Jenkins is an open source automation server written in Java. It is used to automate software development process via continuous integration and facilitates continuous delivery.

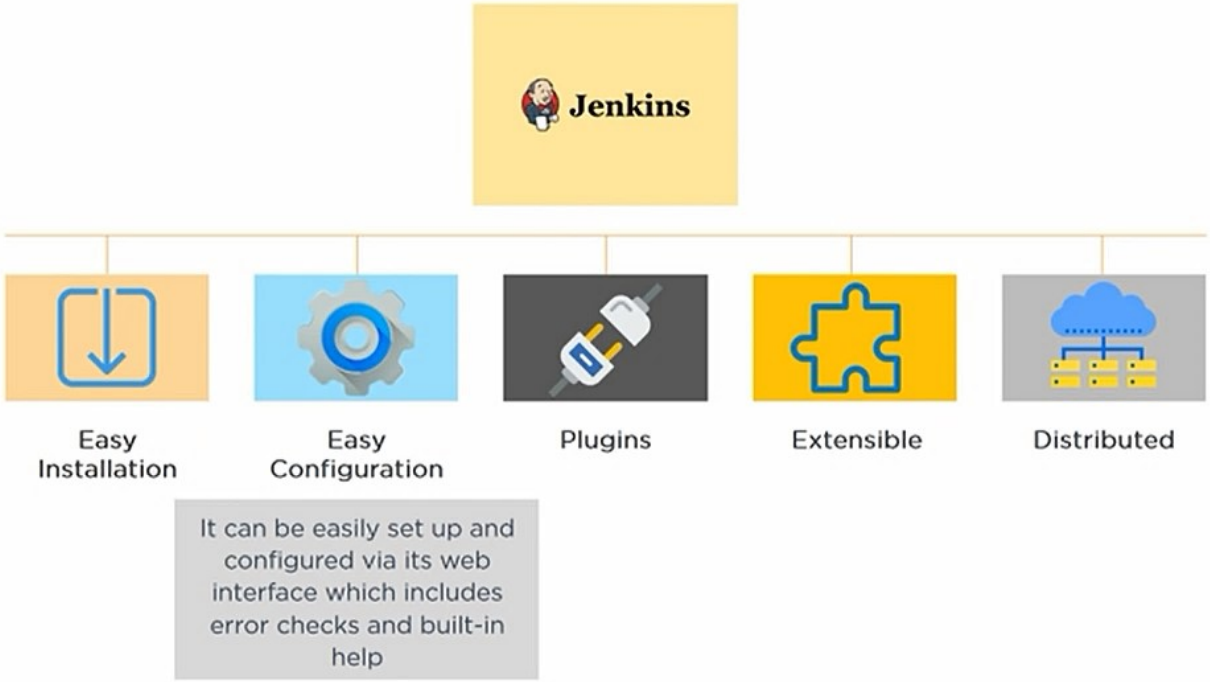



Features of Jenkins

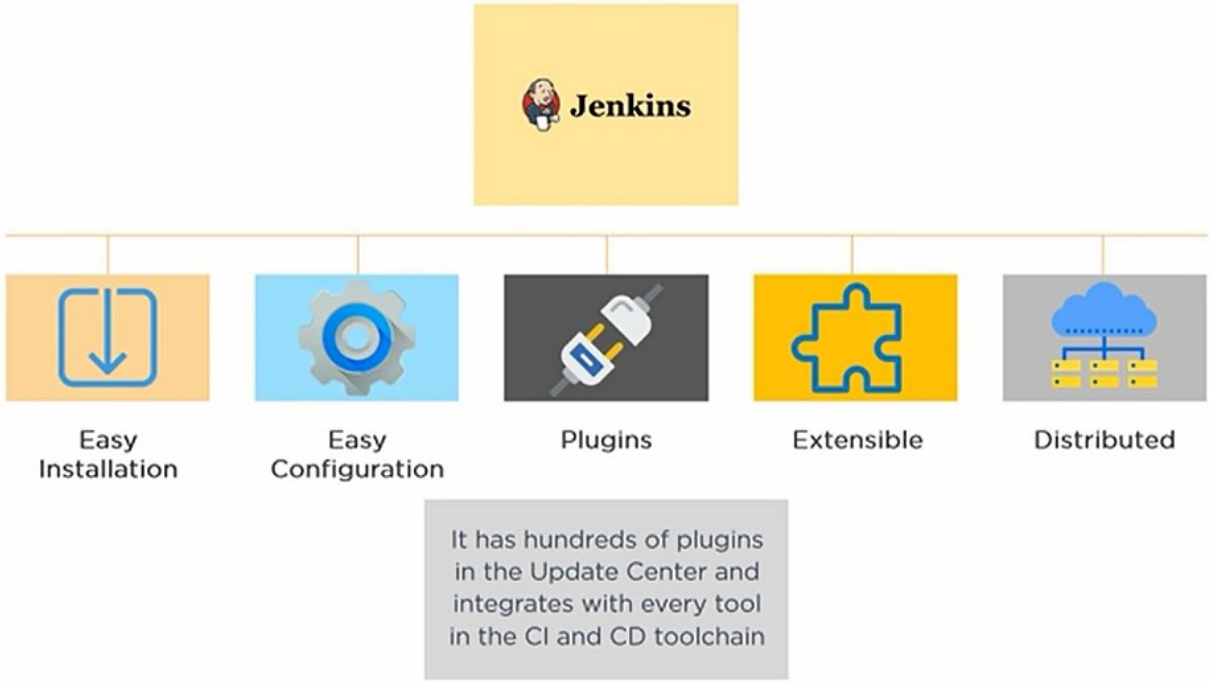
Features of Jenkins



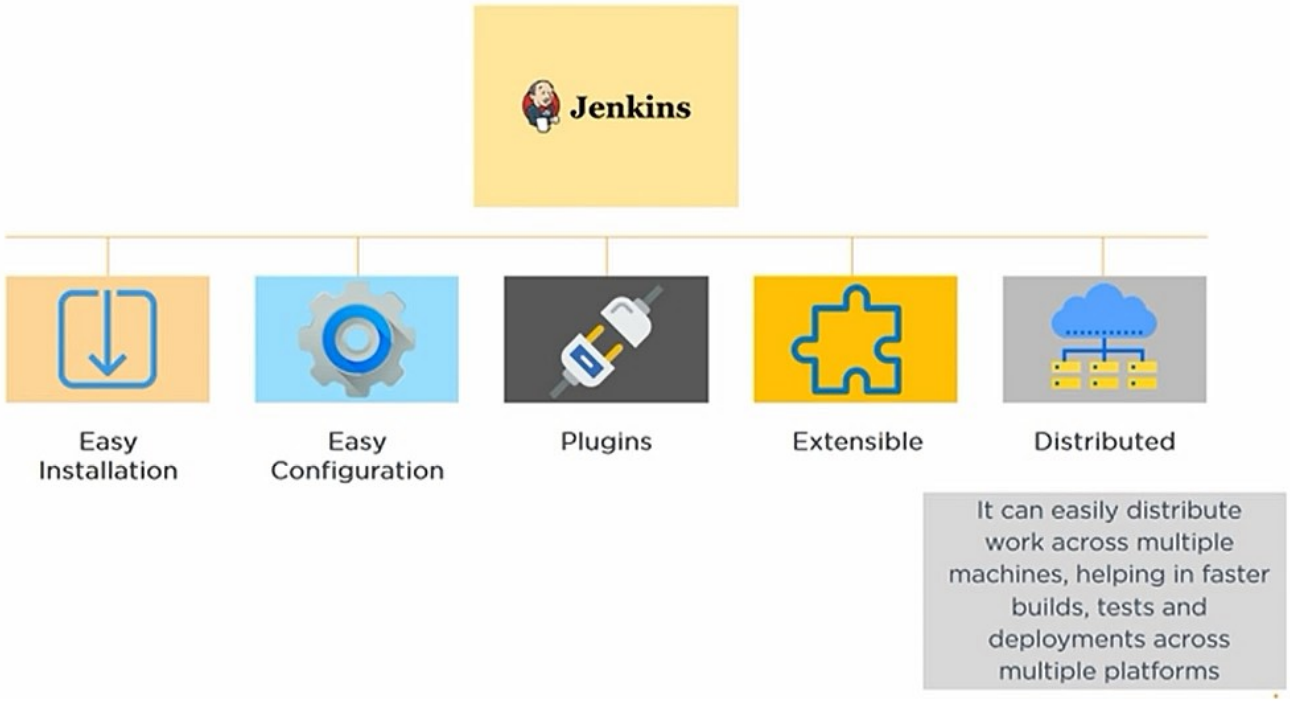
Features of Jenkins



Features of Jenkins

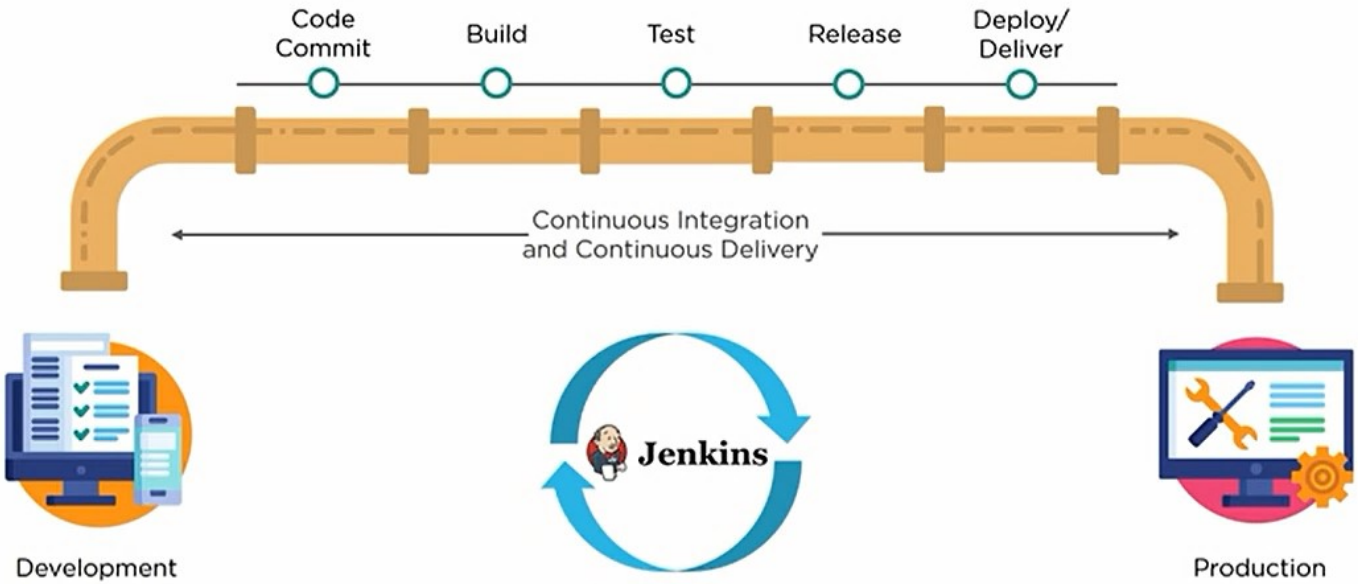


Features of Jenkins



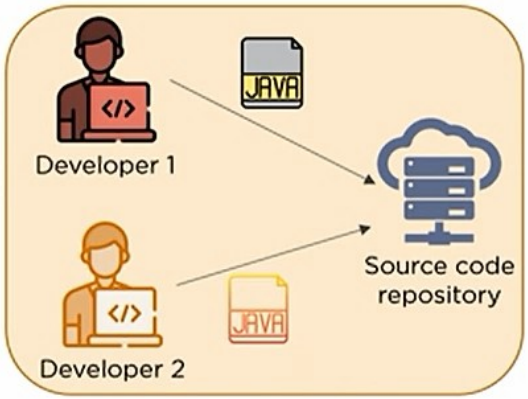

Jenkins Pipeline

Jenkins Pipeline



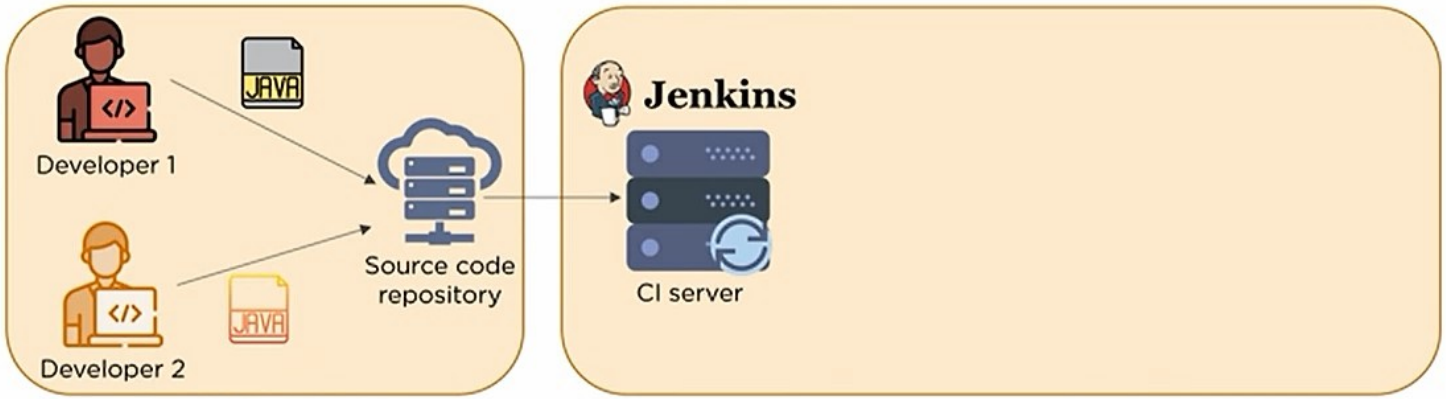

Jenkins Architecture

Jenkins Architecture



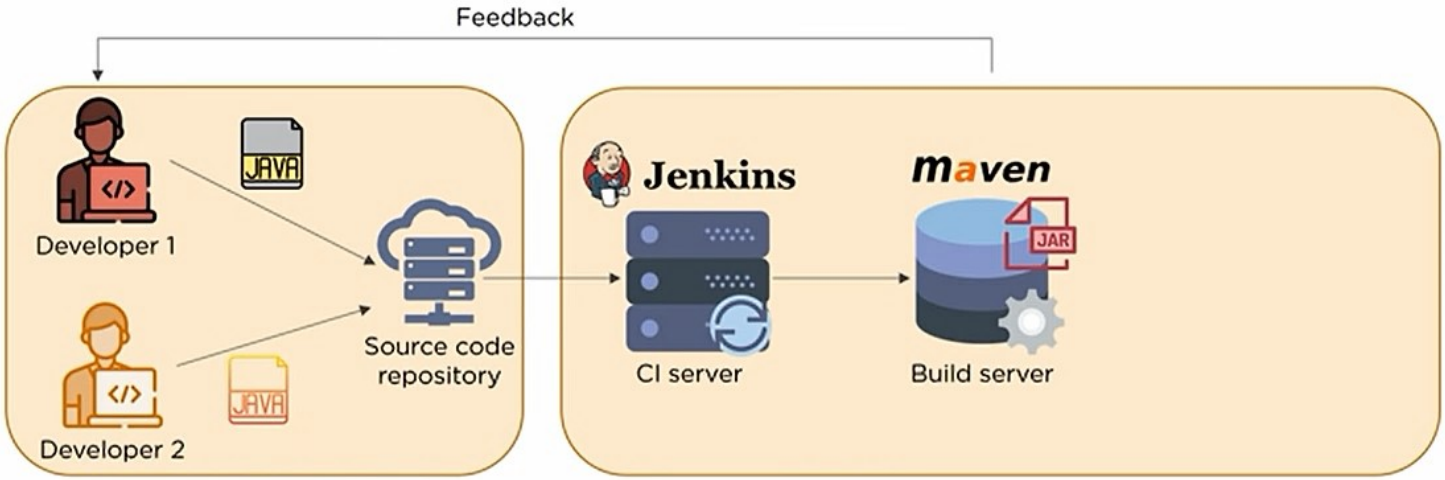
Developers commit changes to the source code

Jenkins Architecture



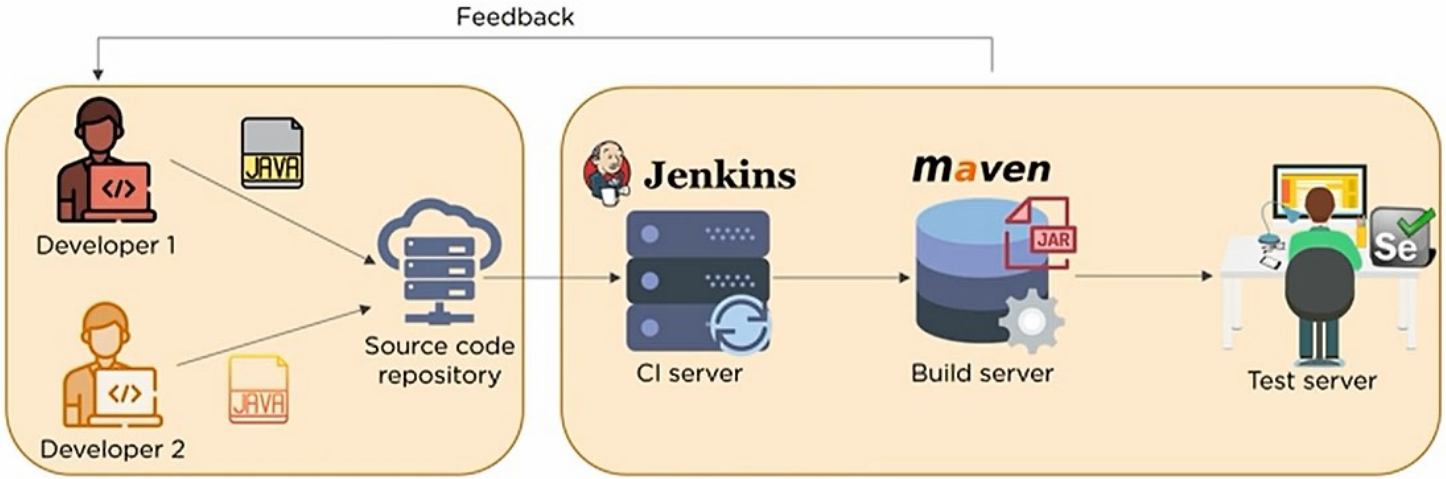
Jenkins server checks the repository at regular intervals and pulls any newly available code

Jenkins Architecture



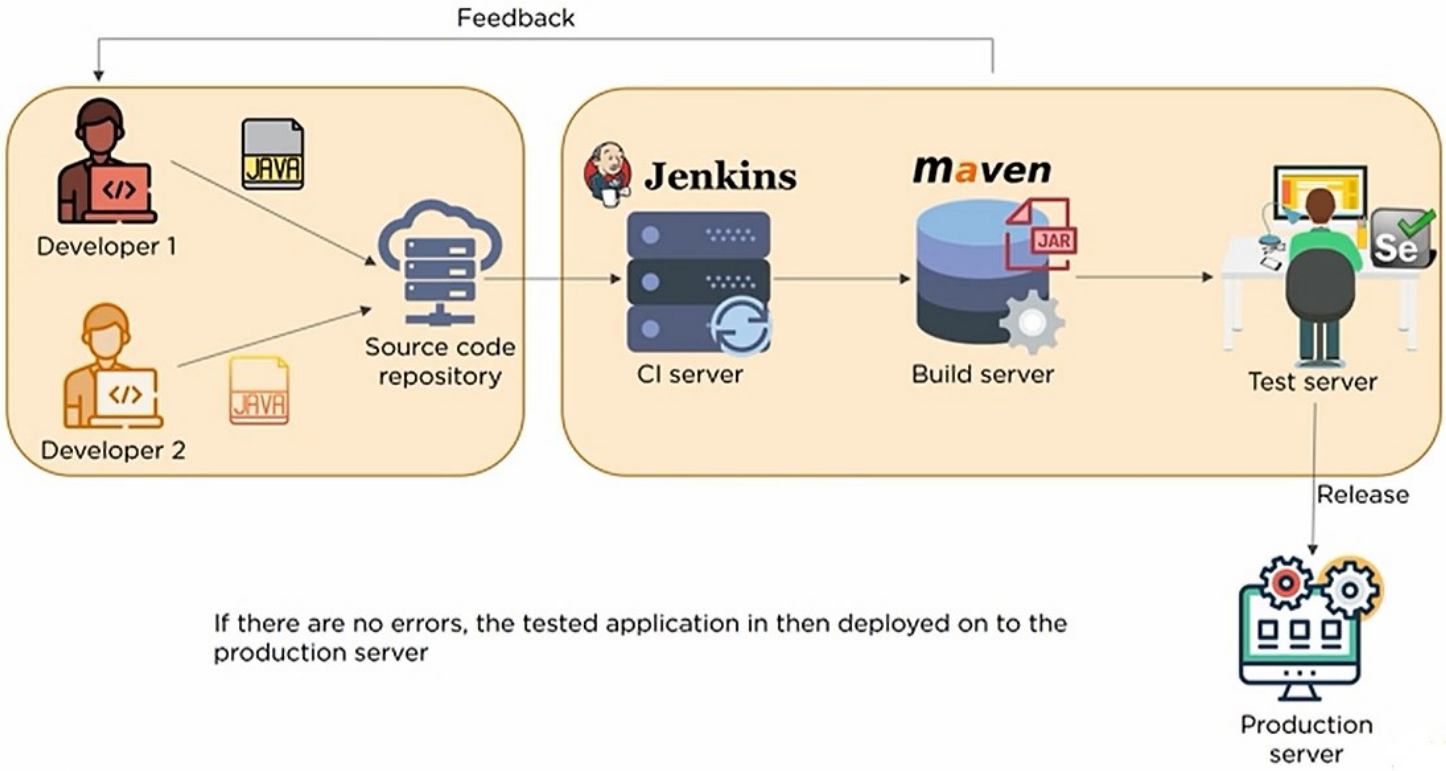
Build Server builds the code into an executable file. Here, Java file is being converted into JAR file. In case the build fails, a feedback is sent to the developers

Jenkins Architecture



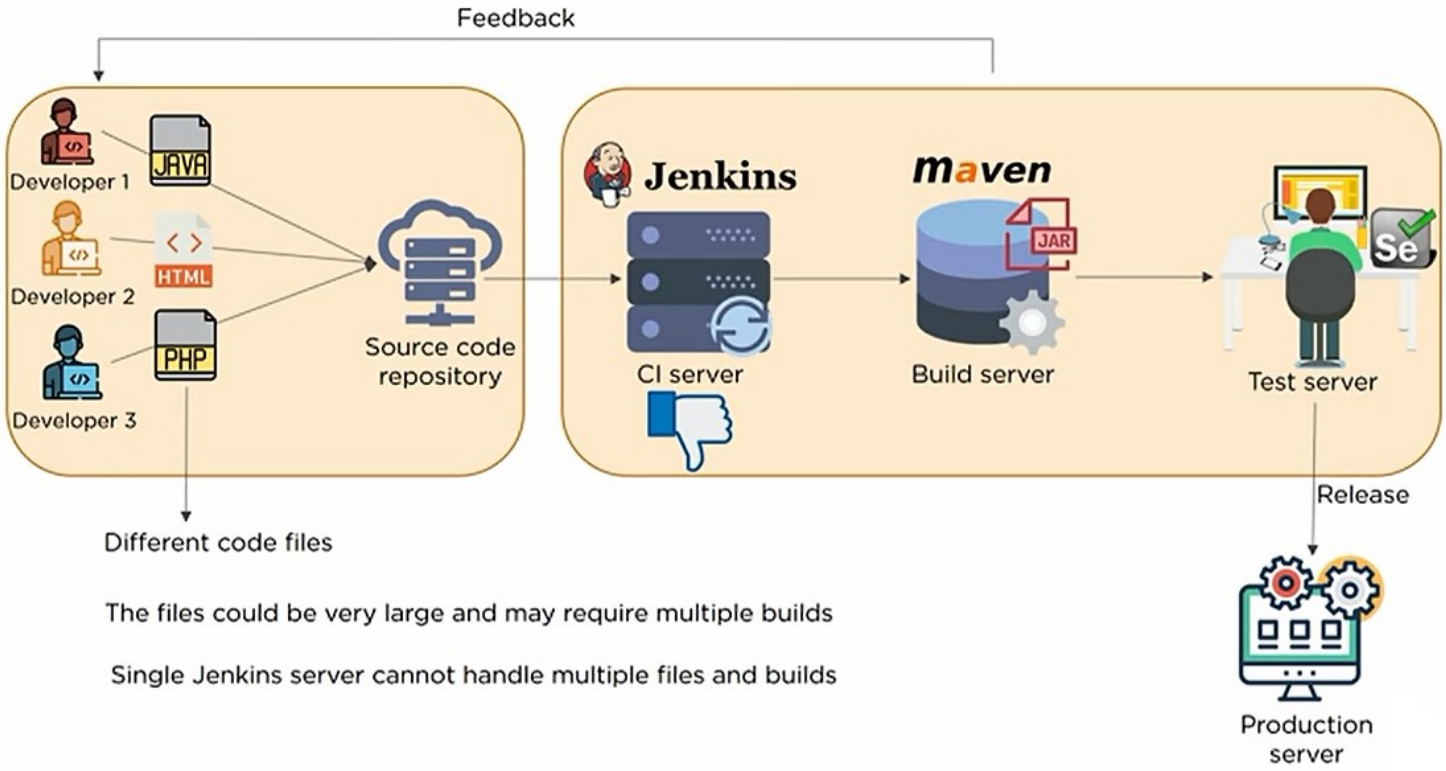
Jenkins then deploys the build application on to test server for testing. If the test fails, feedback is immediately passed on to the developers

Jenkins Architecture



If there are no errors, the tested application in then deployed on to the production server

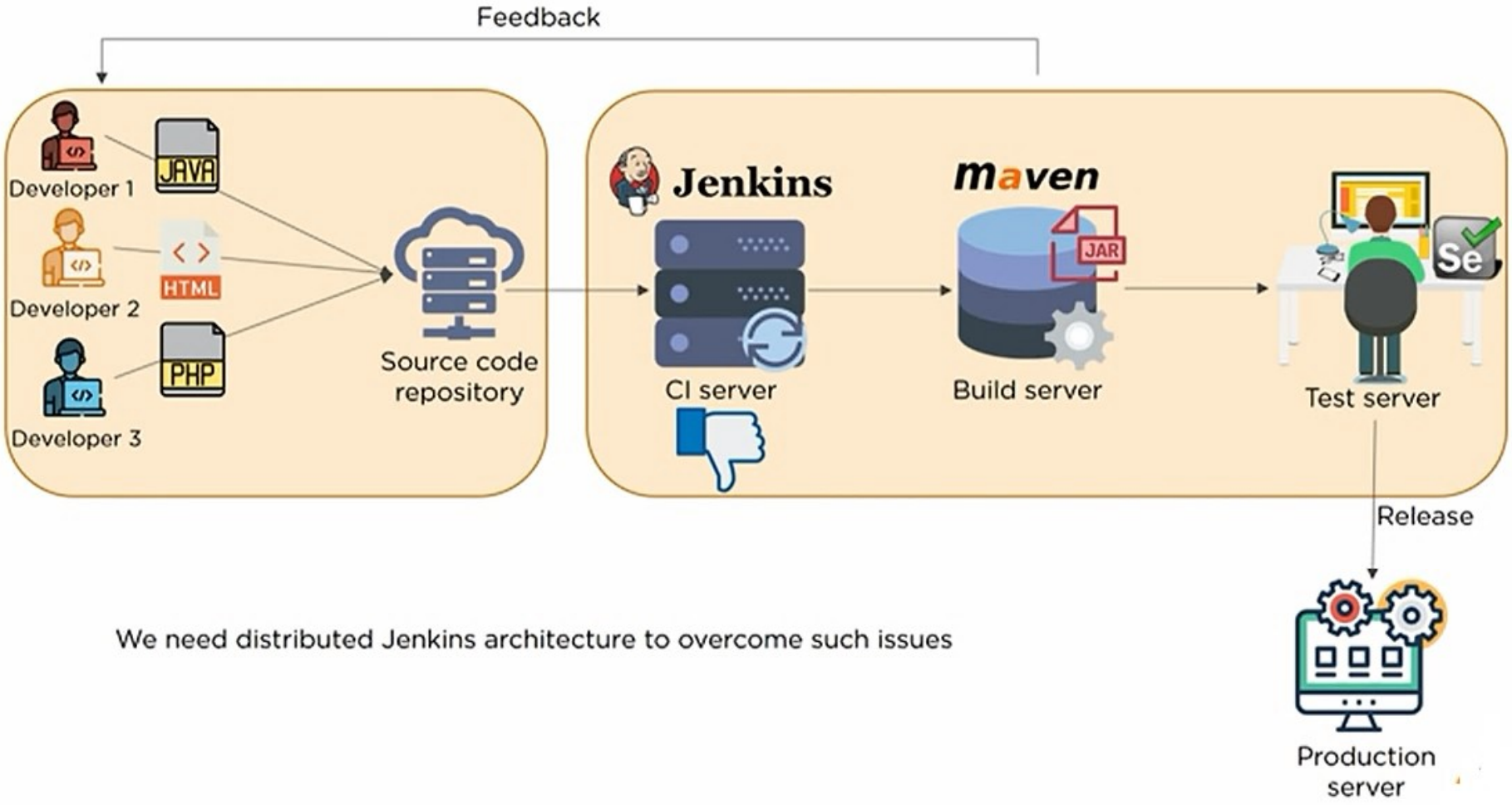
Jenkins Architecture



The files could be very large and may require multiple builds

Single Jenkins server cannot handle multiple files and builds

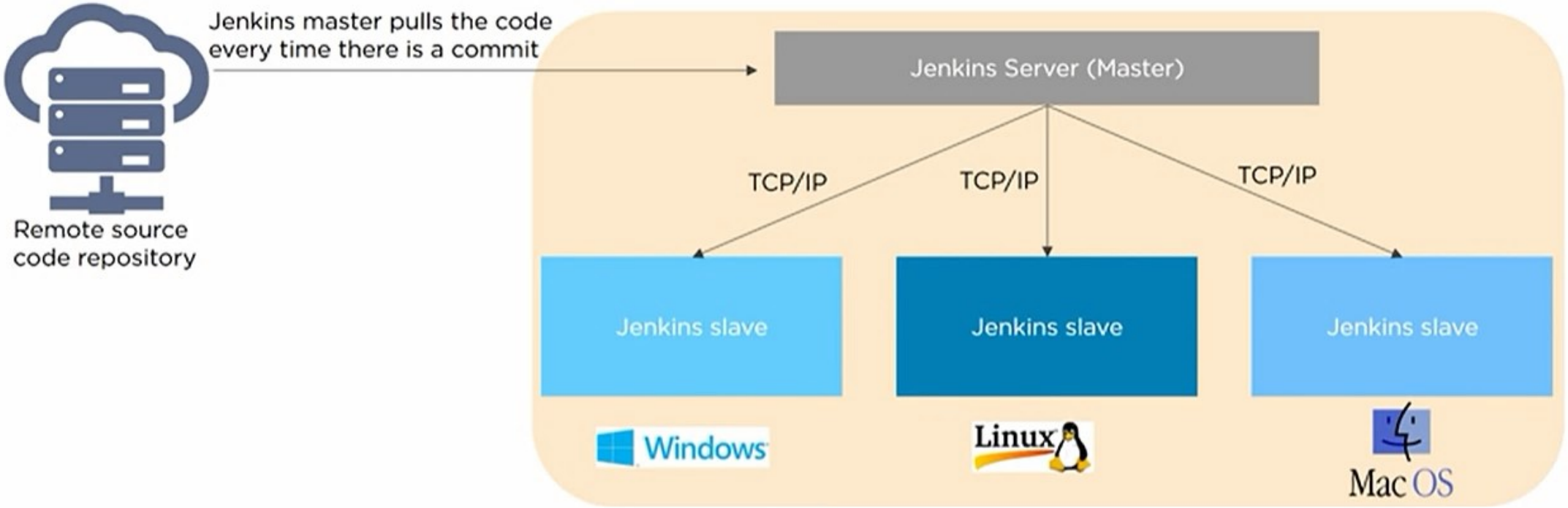
Jenkins Architecture



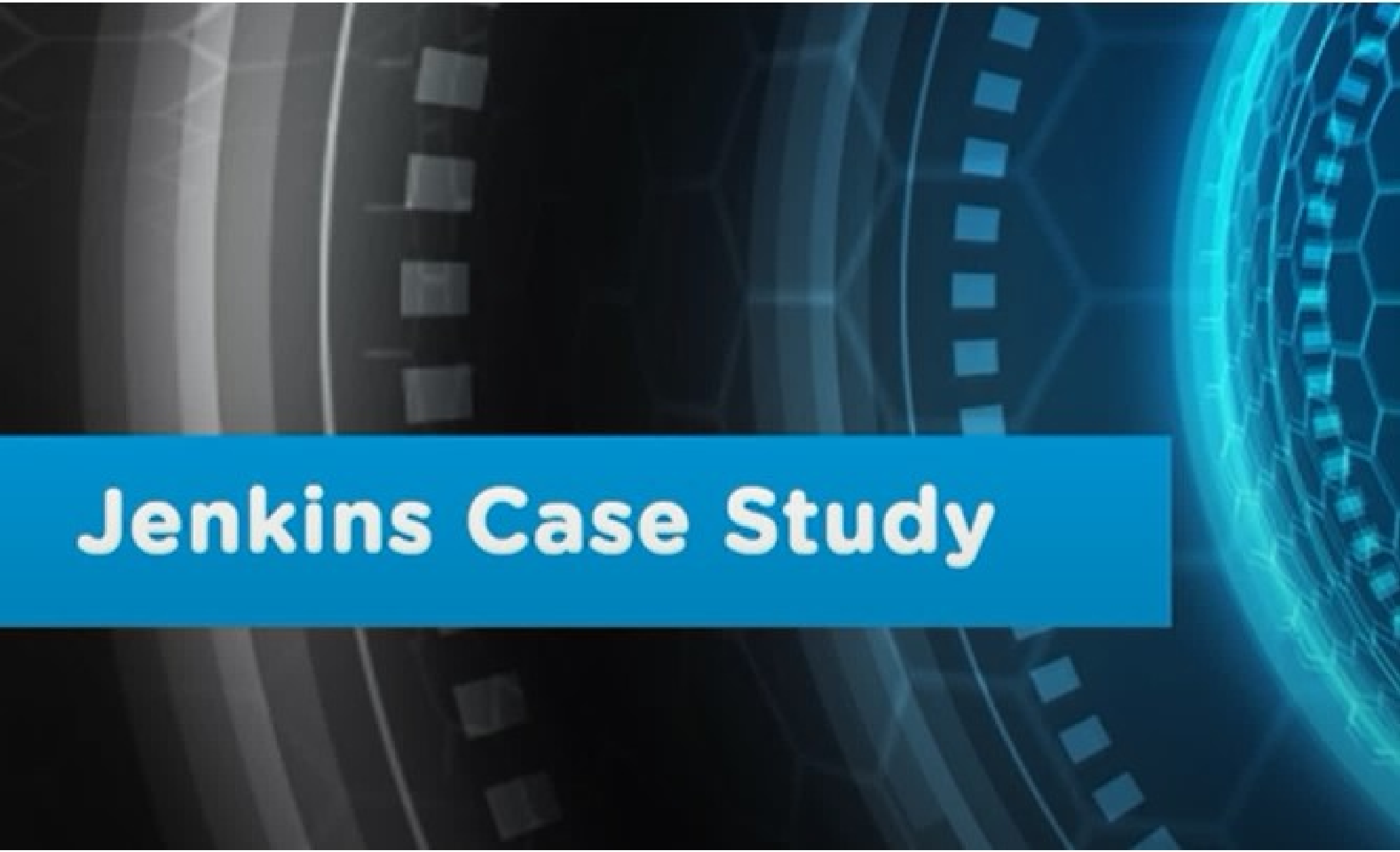
We need distributed Jenkins architecture to overcome such issues


Master - Slave
Architecture

Jenkins Master-Slave Architecture



- Jenkins master distributes its workload to all the slaves
- On request from Jenkins master, the slaves carry out builds and tests and produce test reports


Jenkins Case Study

Jenkins Case Study

Automotive systems are becoming more sophisticated and complex, so the focus for automotive manufacturers is shifting from hardware to software

Many vehicle features, capabilities and performance enhancements are being implemented in software

Jenkins Case Study

BOSCH

BOSCH found a growing need to help its software engineers produce and deliver higher quality software faster

CHALLANGE

Manage and streamline the development of increasingly complex automotive software by adopting CI and CD practices to shorten the entire development and delivery process

Jenkins Case Study

BOSCH

CloudBees
The Enterprise Jenkins Company

It helped them improve development efficiency by increasing automation, stability and transparency

Jenkins Case Study

BOSCH

CloudBees
The Enterprise Jenkins Company

RESULTS

- 3 day build process reduced to less than 3 hours
- Large scale deployment kept on track by expert support
- Visibility and transparency improved with Jenkins Operations support

Key Takeaways