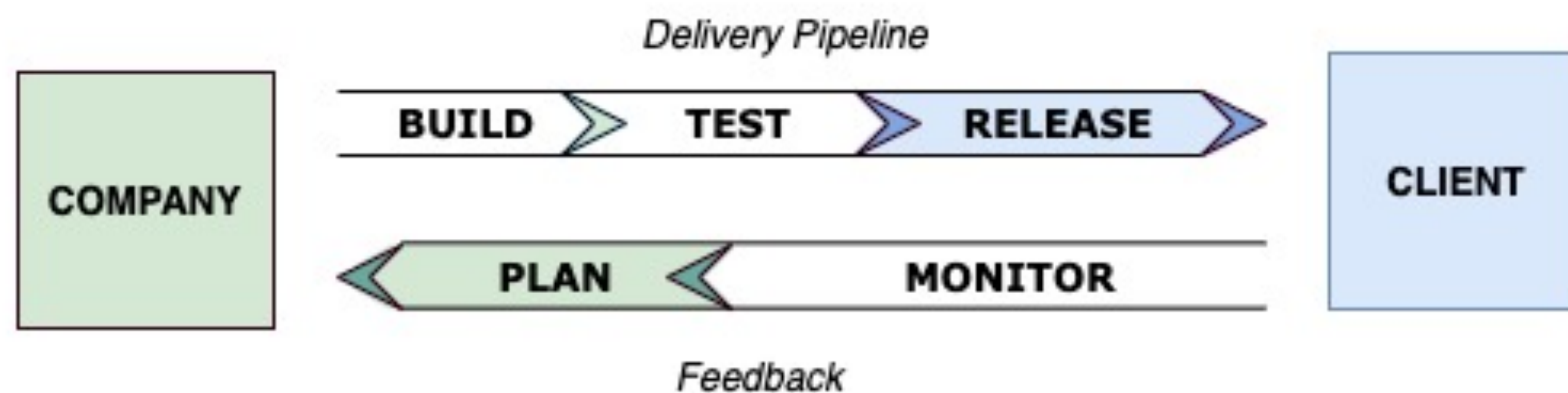


DEVOPS

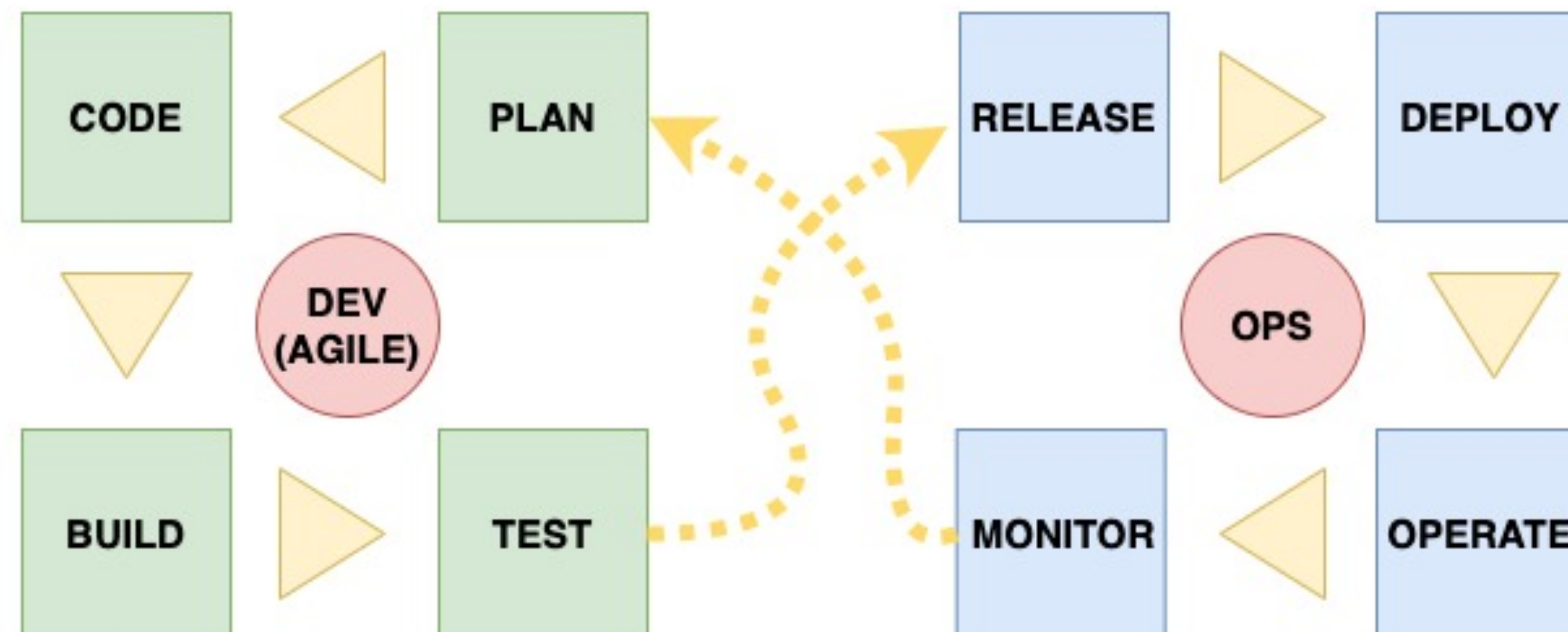
◆ ŠTA JE DEVOPS?

- **Development and Operations**
- Princip u modernoj metodologiji razvoja softvera koji se zasniva na puštanju u rad manjih promena na aplikaciji kako bi se što više smanjio uticaj grešaka na krajnje korisnike, za razliku od velikih *deployment*-a koji prave dosta promena koje mogu da dovedu do velikih problema



◆ ŠTA JE DEVOPS?

- Kombinuje različite alate i prakse kako bi kompanija bila u mogućnosti da isporučuje softverska rešenja brže u odnosu na tradicionalni model isporuke softvera
- Prema DevOps modelu, razvojni i operativni timovi sarađuju tokom čitavog životnog ciklusa softvera, od razvoja i testiranja preko *deployment*-a i puštanja u rad





TRADICIONALNI MODEL ISPORUKE VS DEVOPS

4

- Kod tradicionalnog modela, razvojni tim piše i testira kod u svom privatnom okruženju, bez da je potpuno svestan praksi koje operativni timovi praktikuju
- Klijentu se brže isporučuju rešenja, brži odziv na zahteve
- Automatizacijom procesa se smanjuje prostor za greške





PREDNOSTI

5

◆ BRZA ISPORUKA

- Povećanjem frekvencije isporuke manjih celina, proizvod se brže razvija
- Brži odziv na zahteve klijenata i ispravku *bug*-ova
- Korisnici brže dobijaju nove funkcionalnosti i ispravke *bug*-ova
- Prilagođavanje promenama na tržištu

◆ POUZDANOST

- Kontinualnom integracijom i isporukom novih rešenja se povećava verovatnoća da je svaka nova promena funkcionalna i sigurna za upotrebu



PREDNOSTI

6

◆ UNAPREĐENJE SARADNJE

- Razvojni i operativni timovi sarađuju i dele odgovornosti
- Ušteda vremena

◆ BEZBEDNOST

- Bezbednost treba da bude uključena u svaku fazu DevOps životnog ciklusa, uključujući faze dizajna, izgradnje, testiranja, *deploy*-a, održavanja...
- Danas se ovo popularno naziva *DevSecOps* koji ima za cilj unapređenje bezbednosnih aspekata kroz čitav DevOps *workflow*



CI VS CD

7

◆ CONTINUOUS INTEGRATION (CI)

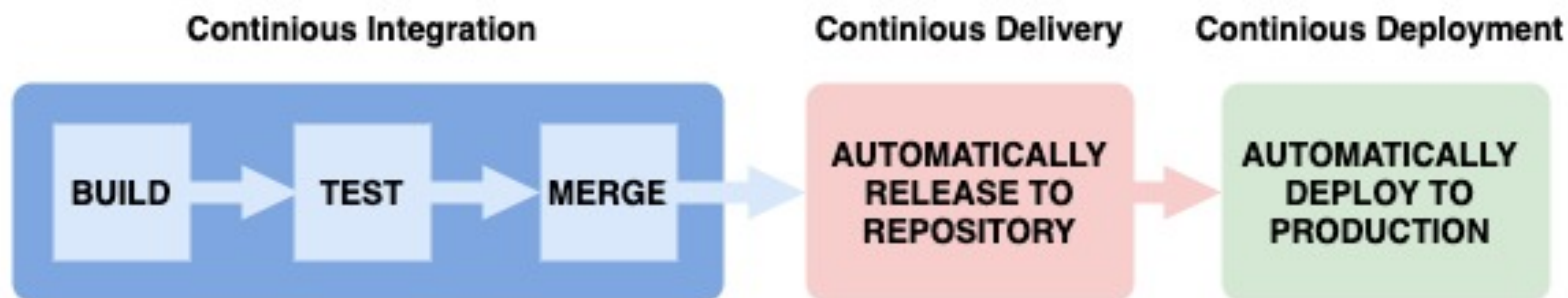
- Praksa razvoja softvera gde se praktikuje redovno spajanje izmena koda na centralni repozitorijum, nakon čega se pokreću automatizovane *build* skripte i testovi
- Cilj je brže pronalaženje i rešavanje *bug*-ova, poboljšanje kvaliteta softvera i smanjenje vremena potrebnog za validaciju i *release* nove verzije

◆ CONTINUOUS DELIVERY (CD)

- Praksa gde se izmene koda automatski *build*-uju, testiraju i pripremaju za puštanje u produkciju
- Proširuje CI praksu tako što se sve izmene direktno puštaju na testno i/ili produkciono okruženje nakon *build* faze

◆ ŠTA JE CI/CD PIPELINE?

- Niz koraka koji se moraju izvršiti da bi se isporučila nova verzija softvera.
- Uvodi automatizaciju za unapređenje procesa razvoja aplikacija, posebno u fazama integracije, testiranja i isporuke.

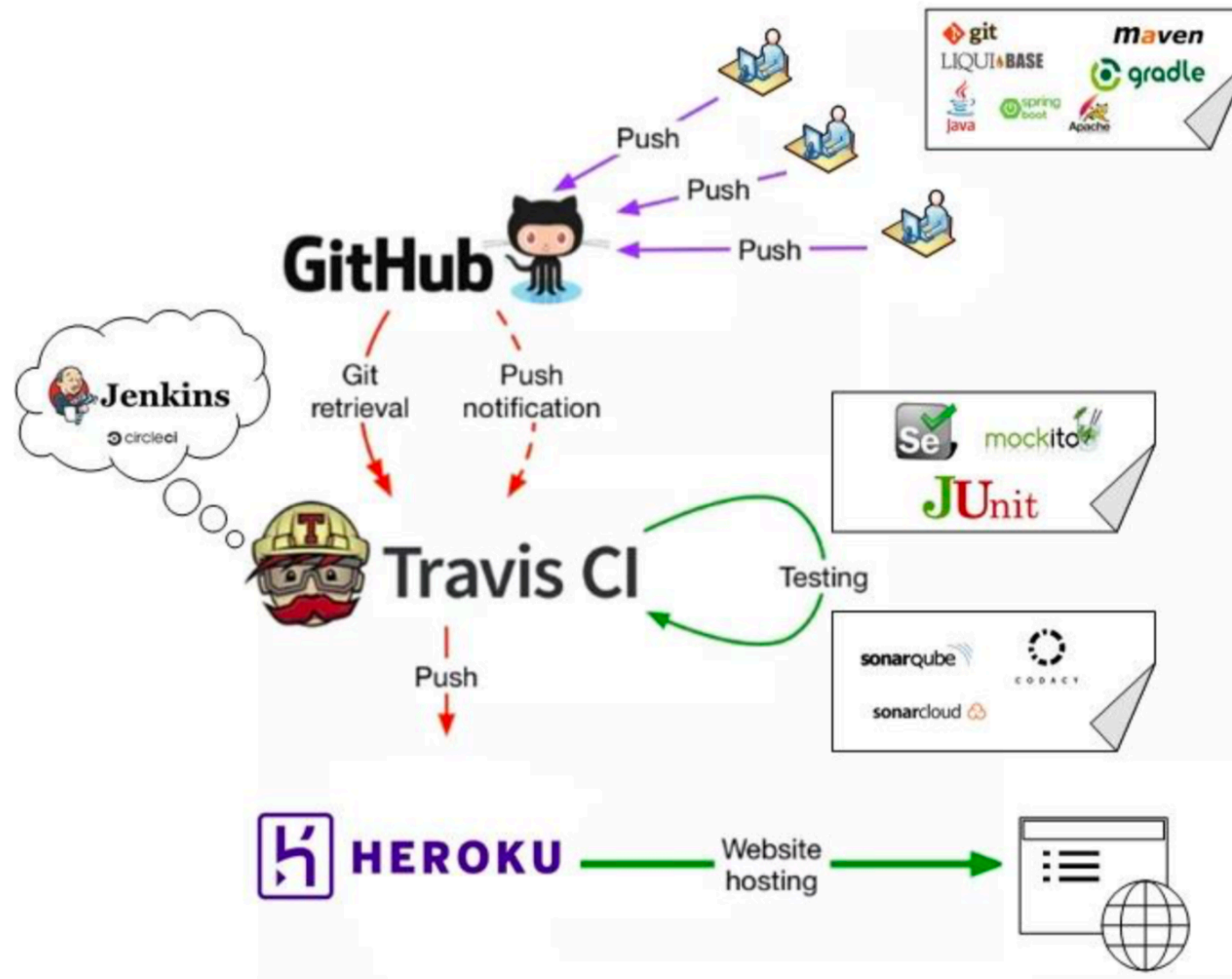




TOOLS

Source Code control	CI/CD	Containers
GitHub GitLab Bitbucket	Travis Jenkins Circle CI	Docker Kubernetes
Cloud computing and storage	Monitoring, alerting and incident response tools	IT ticketing
AWS Azure Google Cloud	New Relic Datadog Instana	Jira Trello HubSpot
Communication and collaboration	Documentation	Testing
Slack Microsoft Teams Rocket Chat	Confluence Slite	Selenium Gremlin

PRIMER





TRAVIS CI

11

◆ ŠTA JE TRAVIS CI?

- *Continuous Integration* platforma
- Kod se *build*-uje i testira na potpuno novom, neutralnom okruženju
- *Feedback* da li je ciklus uspešno završen

◆ ALTERNATIVE

- Jenkins
- CircleCI
- Semaphore CI
- ...



◆ ŠTA JE SONAR?

- Platforma koja omogućava kontinualnu kontrolu kvaliteta koda
- Statička analiza koda za otkrivanje grešaka u kodu, *code smells* i bezbednosnih propusta

◆ ZAŠTO JE STATIČKA ANALIZA KODA VAŽNA?

- Analiza koda bez njegovog izvršavanja
- Detekcija problema u ranim fazama razvoja softvera
- Dodavanje novih pravila kako bi se analiza prilagodila specifičnostima softvera

◆ ALTERNATIVE

- Coverity
- Codacy
- ...



HEROKU

13

◆ ŠTA JE HEROKU?

- Cloud rešenje - Platform as a Service (PaaS)
- Pruža mogućnost *build*-ovanja, isporuke, nadgledanja i skaliranja aplikacija
- Add-ons (Data stores, Caching, Message Queues...)

◆ ALTERNATIVE

- AWS
- Azure
- DigitalOcean
- ...



REFERENCE

14

- ◆ **PRIMER PO UZORU NA** <https://github.com/katarinaa94/deployment-example>
- ◆ **WHAT IS DEVOPS?** <https://aws.amazon.com/devops/what-is-devops/>
- ◆ **WHAT IS DEVOPS?** <https://www.crestdatasys.com/services/devops-services/>
- ◆ **WHAT IS CI/CD PIPELINE?** <https://www.redhat.com/en/topics/devops/what-cicd-pipeline>
- ◆ **TRAVIS CI** <https://docs.travis-ci.com/user/for-beginners/>
- ◆ **SONAR QUBE** <https://docs.sonarqube.org/latest/>
- ◆ **HEROKU** <https://devcenter.heroku.com/categories/reference>

**KOJA SU VAŠA
PITANJA?**