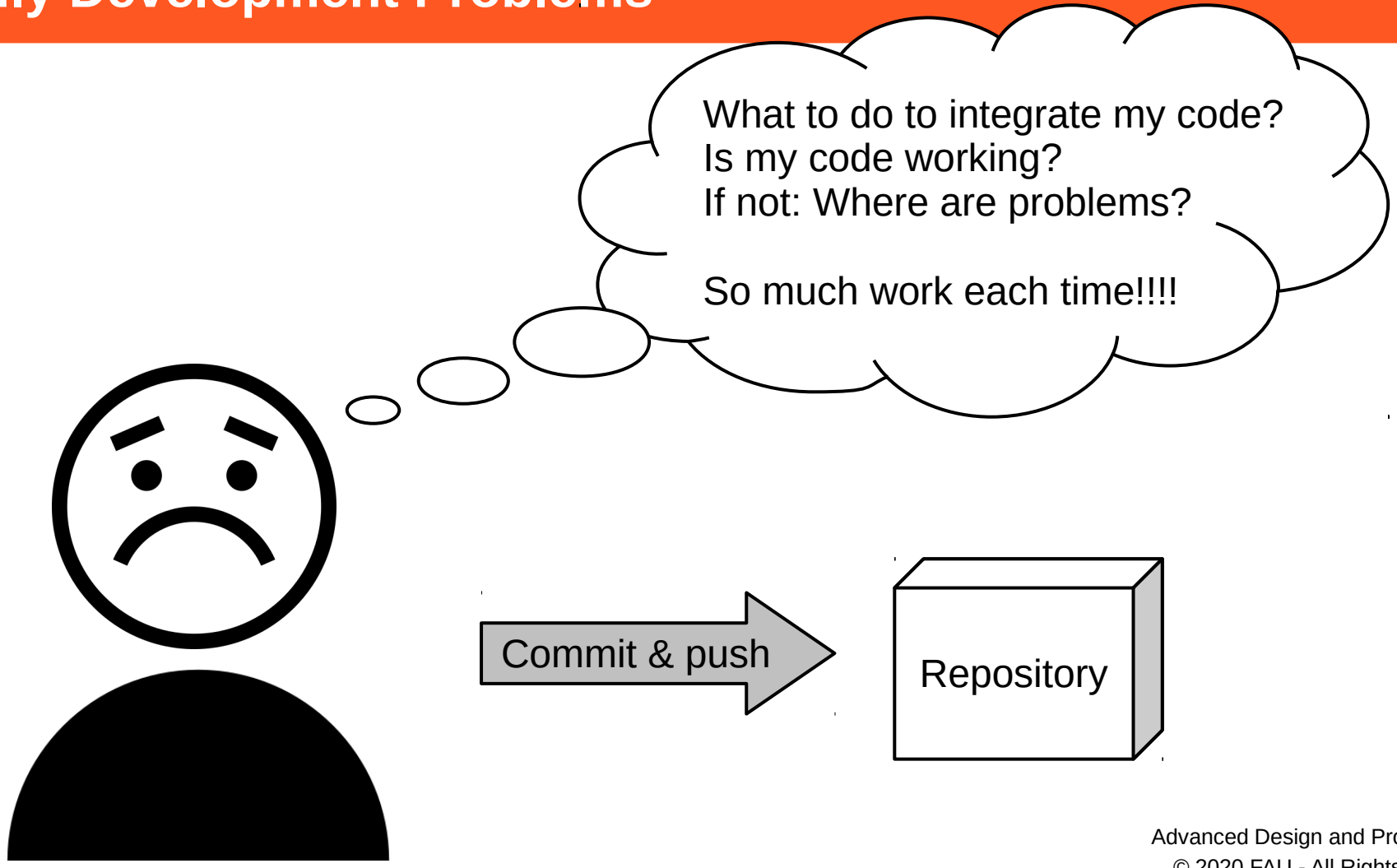
Continuous Integration (CI)

Professorship of Open Source Software

Friedrich-Alexander University Erlangen-Nürnberg

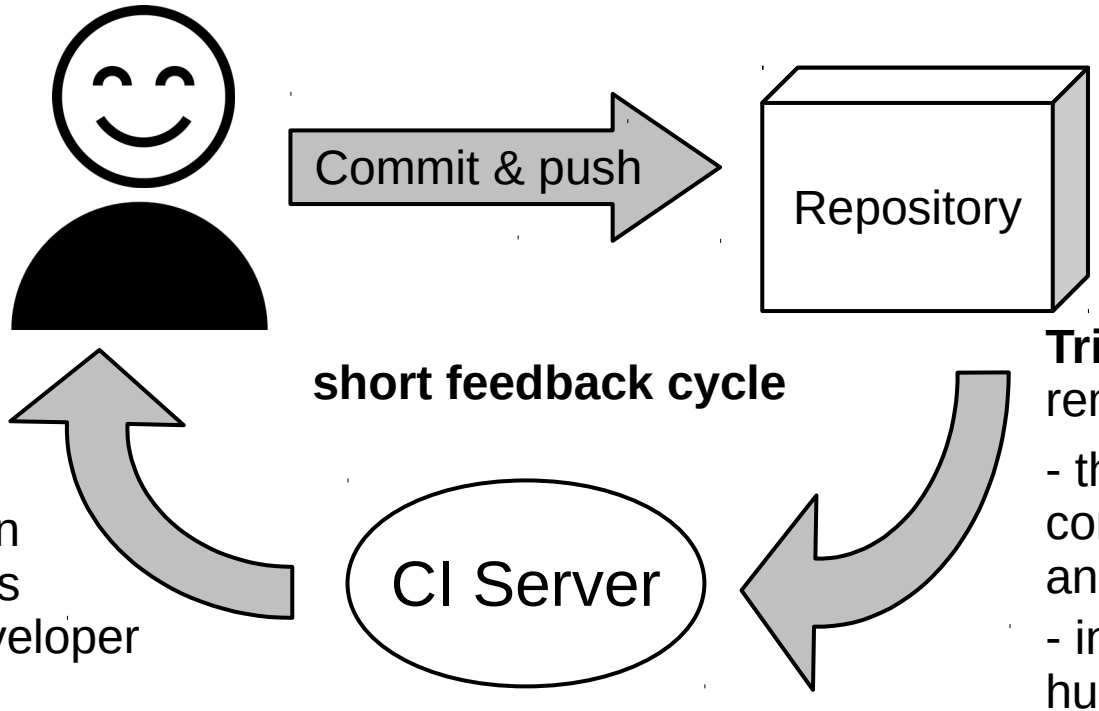
ADAP B02

Daily Development Problems



Continuous Integration (CI)

"Continuous Integration is a software development practice where members of a team integrate their work frequently […] this approach leads to significantly reduced integration problems and allows a team to develop cohesive software more rapidly." [1]



Triggers CI after commit to remote repository

- the system under construction is fetched, built, and tested
- in a fully automated way (no human intervention)

Feedback upon system status is provided to developer and manager

[1] Martin Fowler, https://www.martinfowler.com/articles/continuousIntegration.html

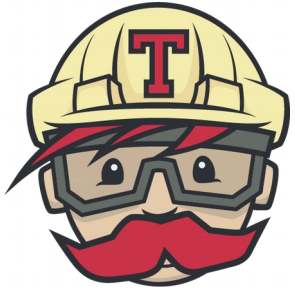
Advantages of CI

- Always know if your project is in a healthy state
- Faster integration
- Faster feedback
- Easier to localize bugs
- Frequent integration of the whole application
- Ideally, improve quality such that you can deploy at any time
- Reduce risks introduced by code changes, e.g. refactoring of the application

CI in practice

- Example Tools:
  - Jenkins (https://jenkins.io/)
  - Travis CI (https://travis-ci.org/)
  - GitLab CI (https://about.gitlab.com/product/continuous-integration/)
  - Buddy (https://buddy.works/)
  - ...

https://wiki.jenkins.io/display/JENKINS/Logo

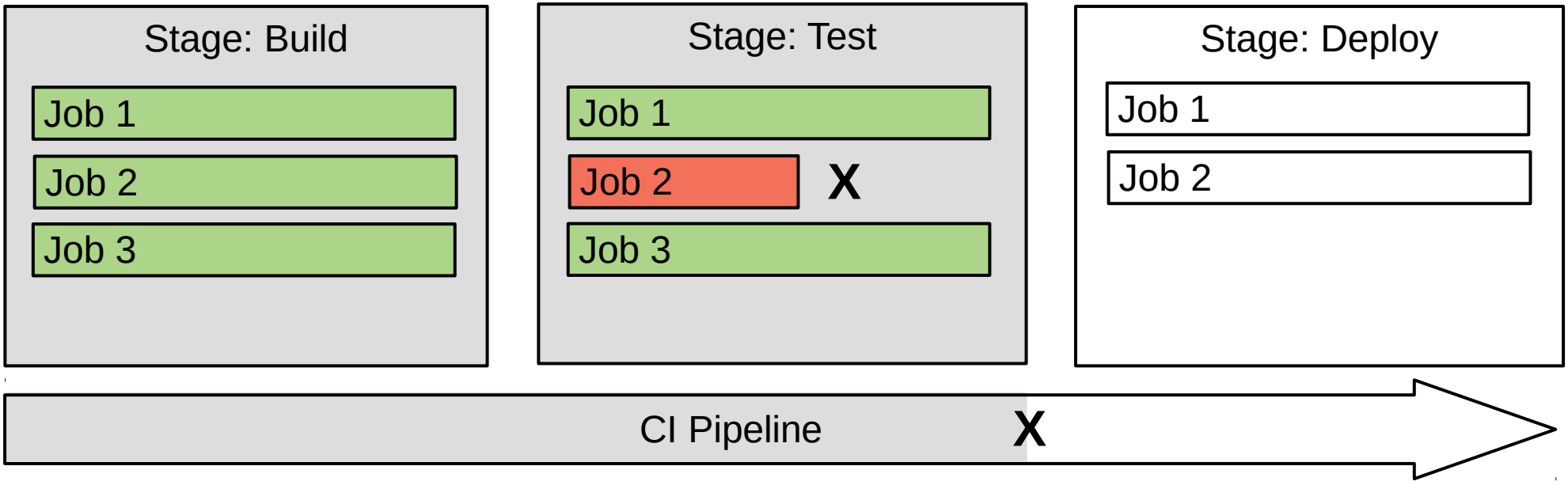


https://travis-ci.com/logo

https://about.gitlab.com/press/press-kit/

CI Pipeline

- Build Stages – group of parallel jobs, stages are run sequentially
- Jobs – executes a task
- Phases – sequential steps of a job (Job Lifecycle)



https://docs.travis-ci.com/user/for-beginners/

Job Lifecycle for Travis CI

.travis.yml

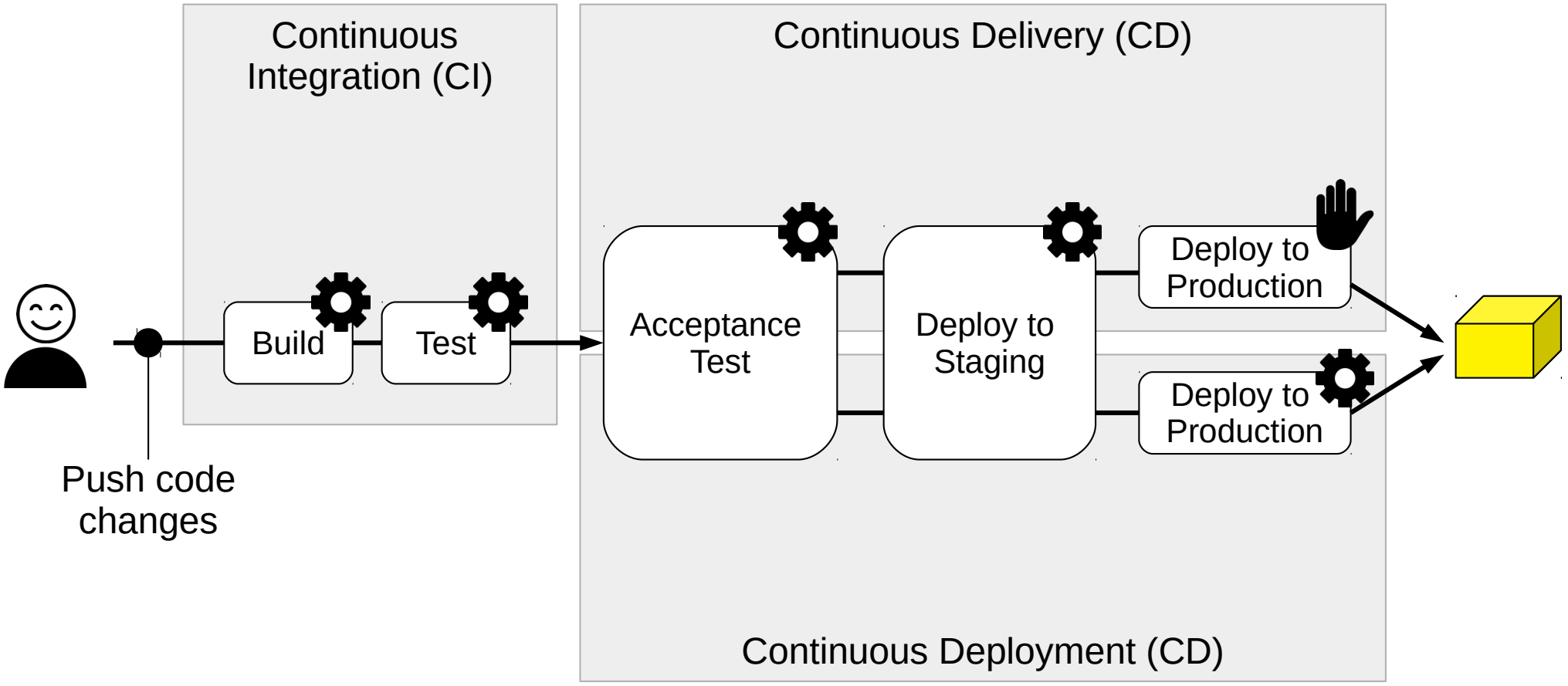
```yaml
language: node_js

before_install: npm config
install: npm install

jobs:
  include:
    - stage: build
      script: buildScript_one.sh
    - stage: build
      script: buildScript_two.sh
    - stage: test
      script: echo "test running"
    - stage: deploy
      script: skip
      deploy:
        provider: npm
        api_key: $NPM_API_KEY
        on: deploy-npm-release
```

more about:
https://docs.travis-ci.com/user/job-lifecycle

1) OPT Install apt addons
2) OPT Install cache components
3) before_install
4) install
5) before_script
6) script
7) OPT before_cache (for cleaning up cache)
8) after_success or after_failure
9) OPT before_deploy
10) OPT deploy
11) OPT after_deploy
12) after_script

main parts

Extensions of Continuous Integreation



Thank you! Questions?

dirk.riehle@fau.de – http://osr.cs.fau.de

dirk@riehle.org – http://dirkriehle.com – @dirkriehle

Credits and License

- Original version
  - © Friedrich-Alexander University Erlangen-Nürnberg, all rights reserved
- Contributions
  - Julia Krause (2019)