

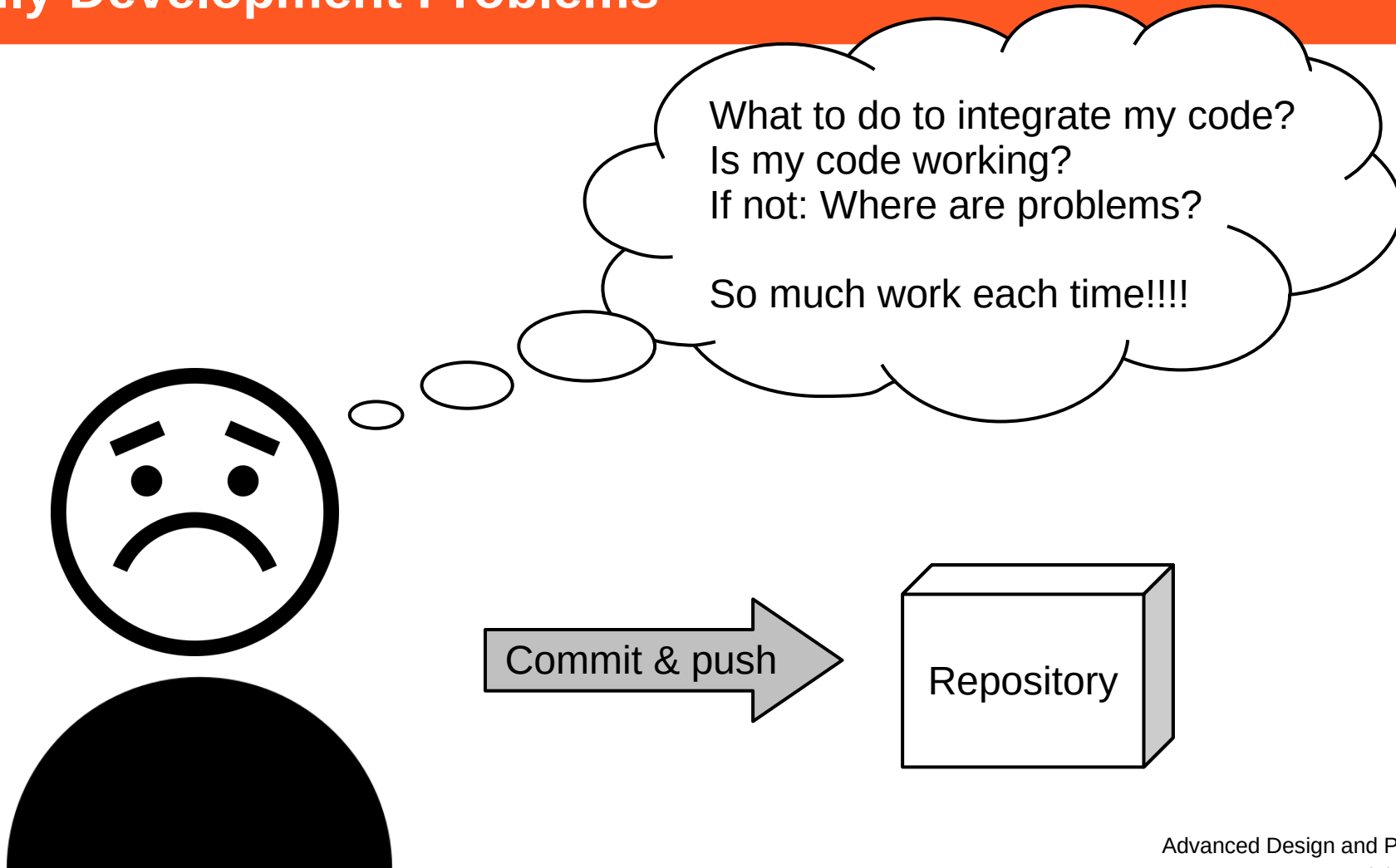
Continuous Integration (CI)

**Professorship of Open Source Software
Friedrich-Alexander University Erlangen-Nürnberg**

ADAP B02

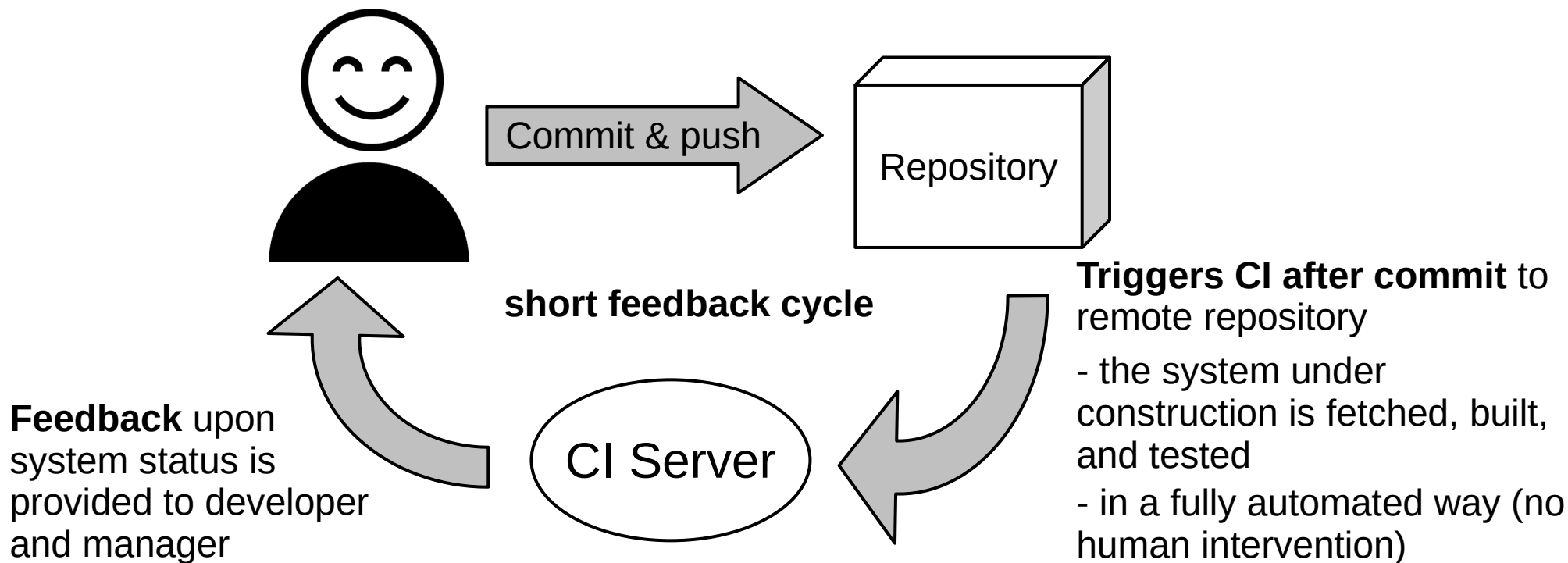
Licensed under [CC BY 4.0 International](#)

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]



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

Advantages of CI

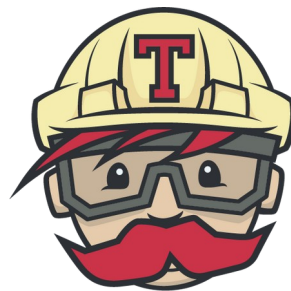
- Always know if your project is in a healthy state
- Faster integration
- Faster feedback
- Easier to localize bugs
- Frequent integration of the whole application
- Ideally, improve quality such that you can deploy at any time
- Reduce risks introduced by code changes, e.g. refactoring of the application

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>



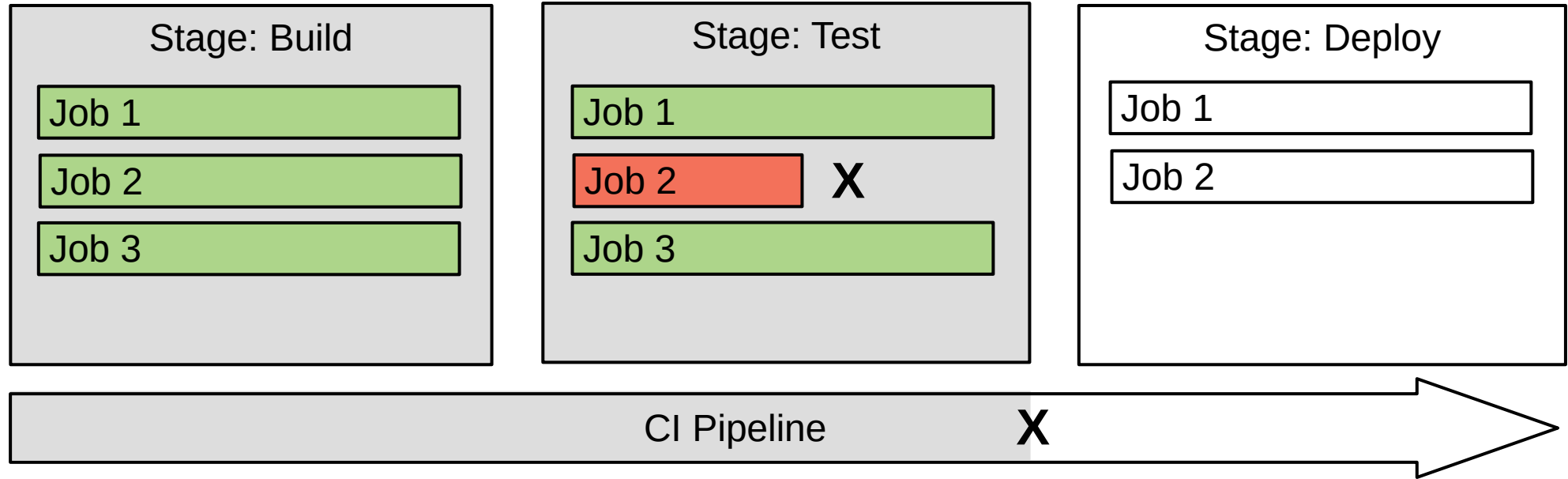
<https://travis-ci.com/logo>



<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)



Job Lifecycle for Travis CI

```
! .travis.yml
1  language: node_js
2
3  before_install: npm config
4  install: npm install
5
6  jobs:
7    include:
8      - stage: build
9        script: buildScript_one.sh
10     - stage: build
11       script: buildScript_two.sh
12     - stage: test
13       script: echo "test running"
14     - stage: deploy
15       script: skip
16     deploy:
17       provider: npm
18       api_key: $NPM_API_KEY
19       on: deploy-npm-release
20
```

.travis.yml

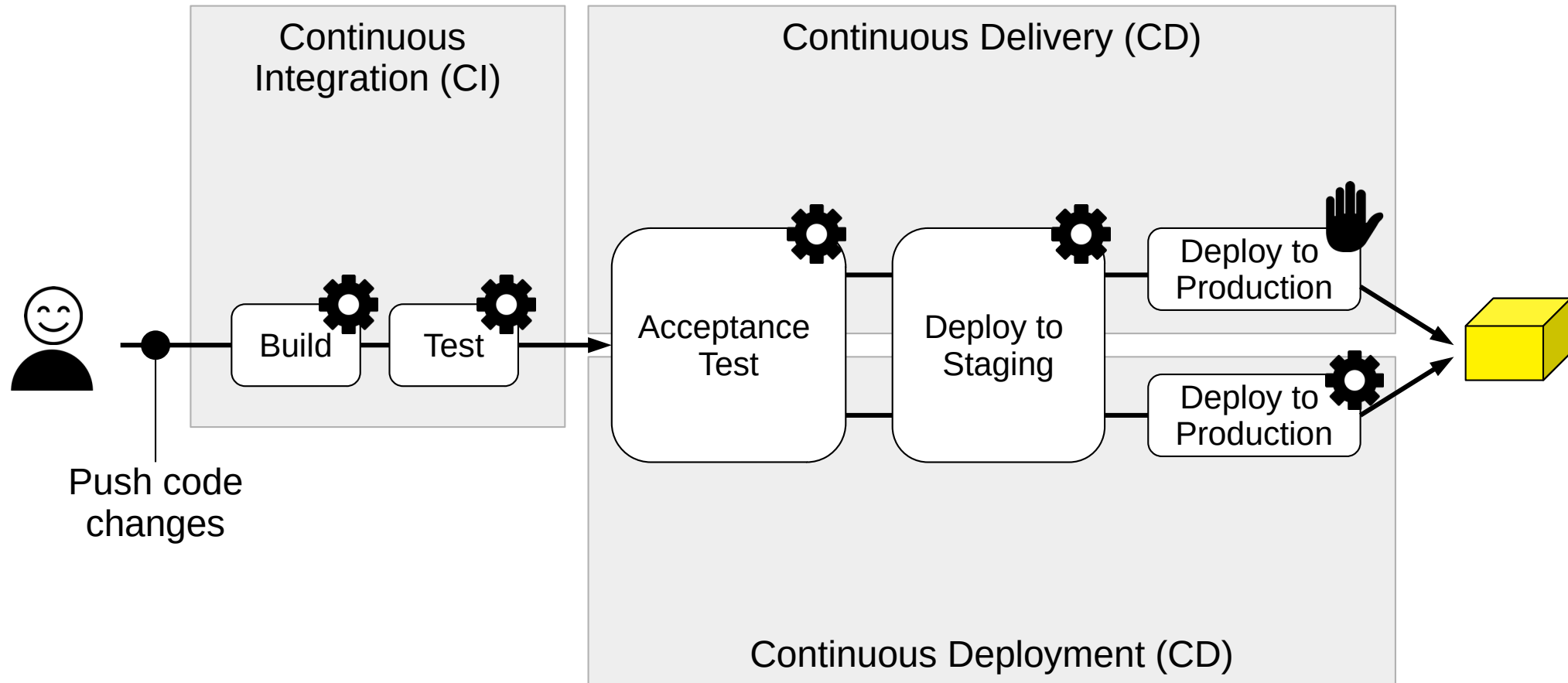
- 1) OPT Install apt addons
- 2) OPT Install cache components
- 3) before_install
- 4) **install** _____
- 5) before_script
- 6) **script** _____
- 7) OPT before_cache (for cleaning up cache)
- 8) after_success or after_failure
- 9) OPT before_deploy
- 10) OPT deploy
- 11) OPT after_deploy
- 12) after_script

main parts

more about:

<https://docs.travis-ci.com/user/job-lifecycle>

Extensions of Continuous Integration



Thank you! Questions?

dirk.riehle@fau.de – <https://oss.cs.fau.de>

dirk@riehle.org – <https://dirkriehle.com> – [@dirkriehle](#)

Legal Notices

- License
 - Licensed under the **CC BY 4.0 International** License
- Copyright
 - © 2019-2021 Friedrich-Alexander University Erlangen-Nürnberg, some rights reserved
- Contributions
 - Julia Krause (2019)