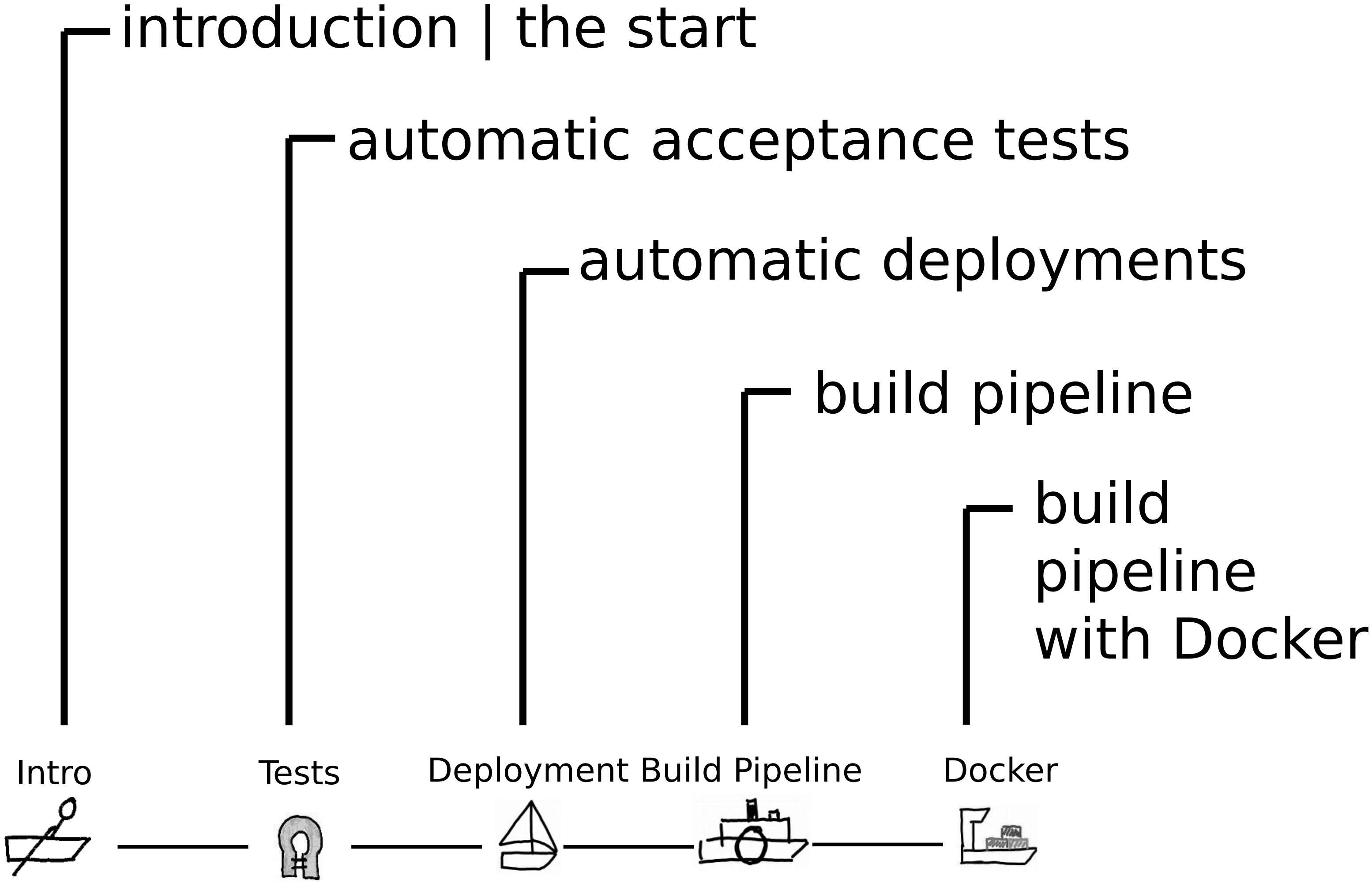
From rowing boats to container ships

a journey to continuous delivery

Tobias Getrost | 1&1 Telecommunication SE

Agenda



introduction | the start

automatic acceptance tests

automatic deployments

build pipeline

build pipeline with Docker

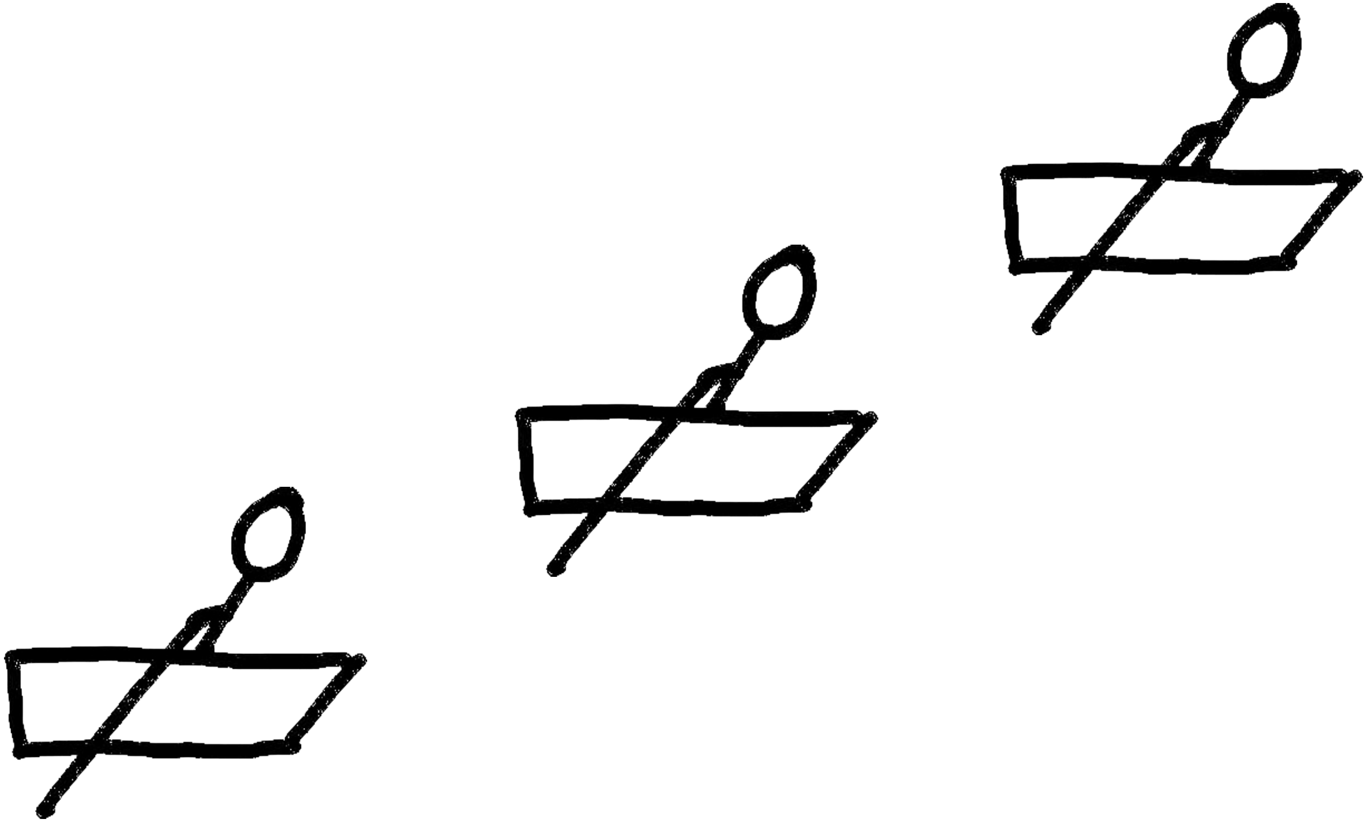
Intro

Tests

Deployment

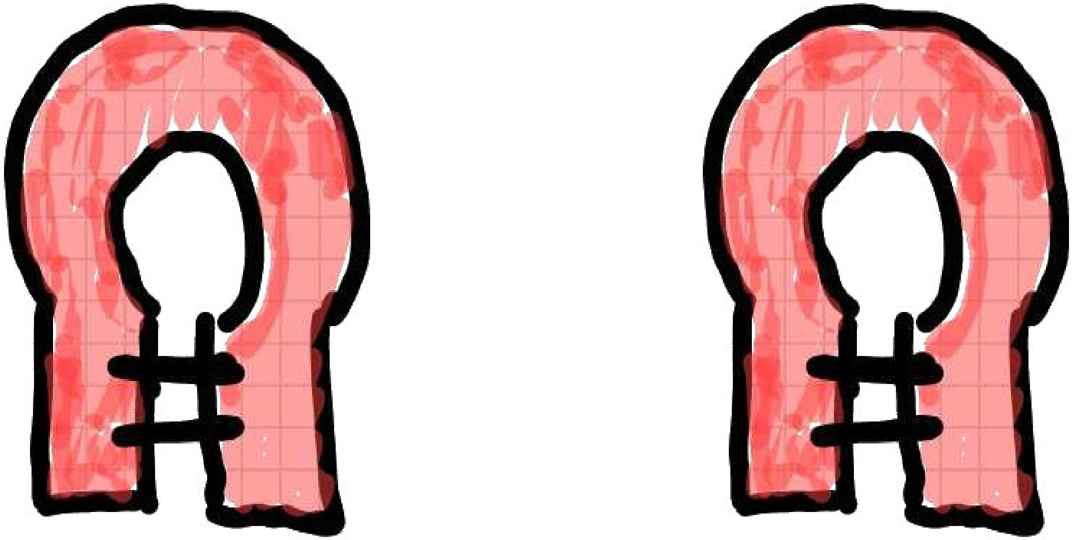
Build Pipeline

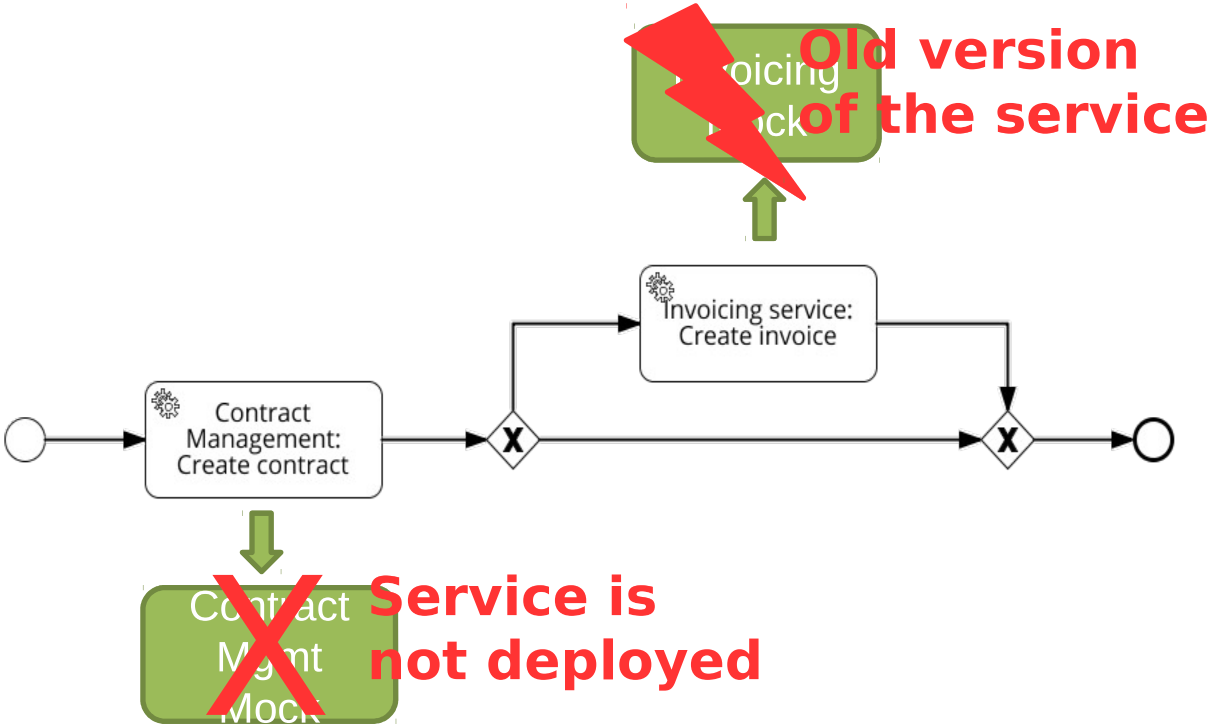
Docker



“Reliable Software Releases through Build, Test and Deployment Automation”

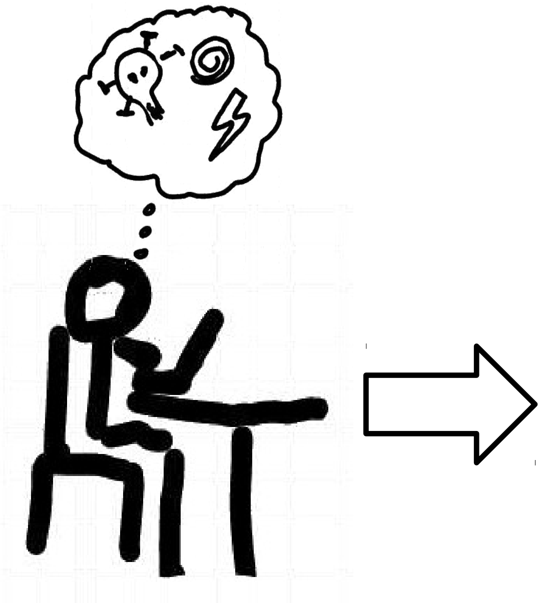
J. Humble, D. Farley: Continuous Delivery

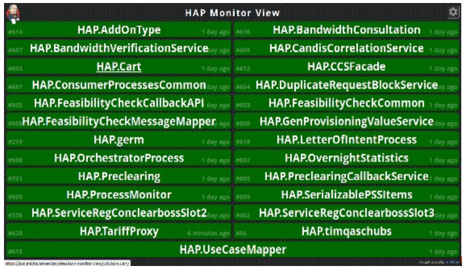



Invoicing Mock
Invoicing service: Create invoice
Contract Management: Create contract
Contract Mgmt Mock

Old version of the service

Service is not deployed

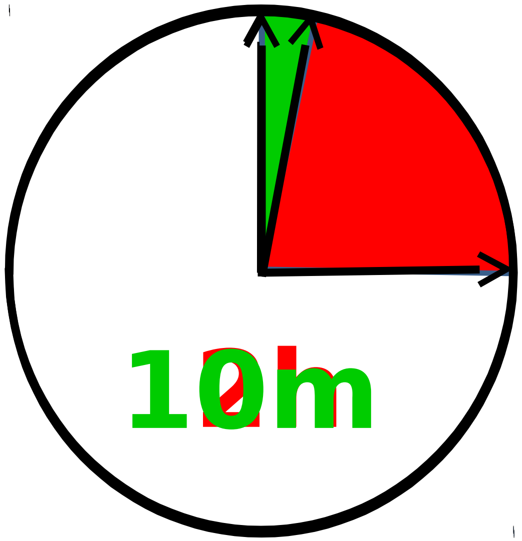


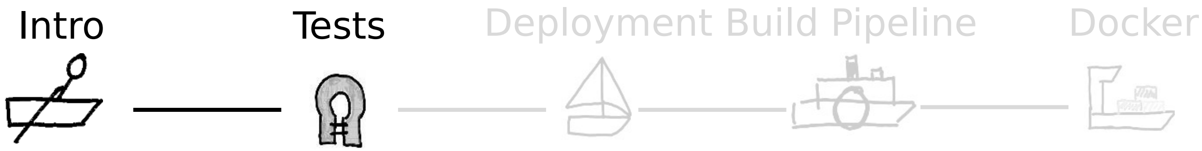


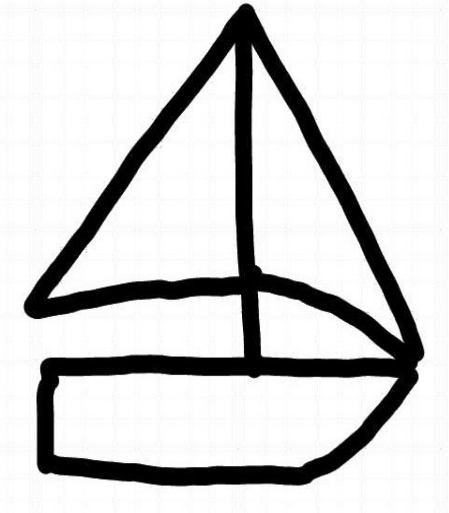
HAP Monitor View

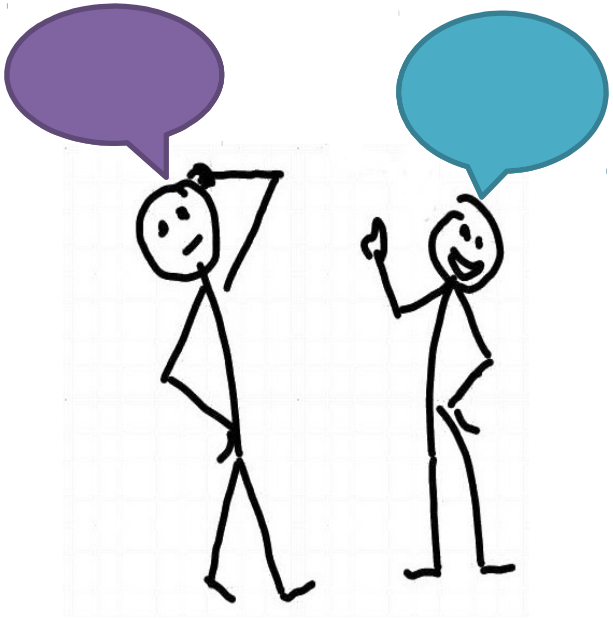
```
#614 HAP.AddOnType 1 day ago                          #616 HAP.BandwidthConsultation 1 day ago
#607 HAP.BandwidthVerificationServiceday ago          #609 HAP.CandisCorrelationService 1 day ago
#693 HAP.Cart 1 day ago                               #612 HAP.CCSFacade 1 day ago
#681 HAP.ConsumerProcessesCommonday ago               #604 HAP.DuplicateRequestBlockServicey ago
#605 HAP.FeasibilityCheckCallbackAPI day ago          #603 HAP.FeasibilityCheckCommon 1 day ago
#608HAP.FeasibilityCheckMessageMappery ago            #608 HAP.GenProvisioningValueServiceday ago
#219 HAP.germ 1 day ago                               #618 HAP.LetterOfIntentProcess 1 day ago
#608 HAP.OrchestratorProcess 1 day ago                #607 HAP.OvernightStatistics 1 day ago
#701 HAP.Preclearing 1 day ago                        #605 HAP.PreclearingCallbackService1 day ago
#605 HAP.ProcessMonitor 1 day ago                     #609 HAP.SerializablePSSItems 1 day ago
#336 HAP.ServiceRegConclearbossSlot2day ago           #462 HAP.ServiceRegConclearbossSlot3day ago
#620 HAP.TariffProxy 6 minutes ago                    #56 HAP.timqaschubs 1 day ago
#618 HAP.UseCaseMapper 1 day ago
```

Intro — Tests — Deployment Build Pipeline — Docker


10m
2h


Intro
Tests
Deployment Build Pipeline
Docker



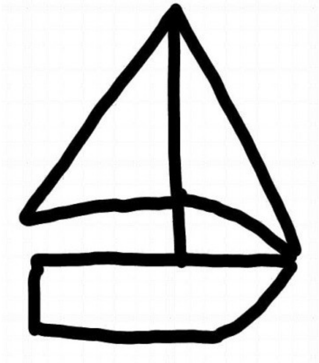


Talk to each other

Automatic deployments

Talk to the stakeholders early
e.g. IT team

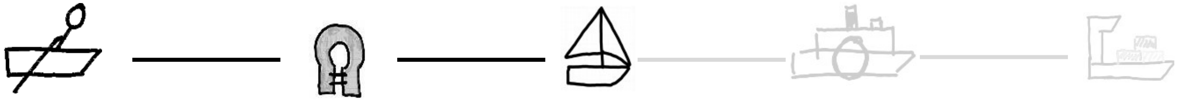
Use the same tools for ALL stages



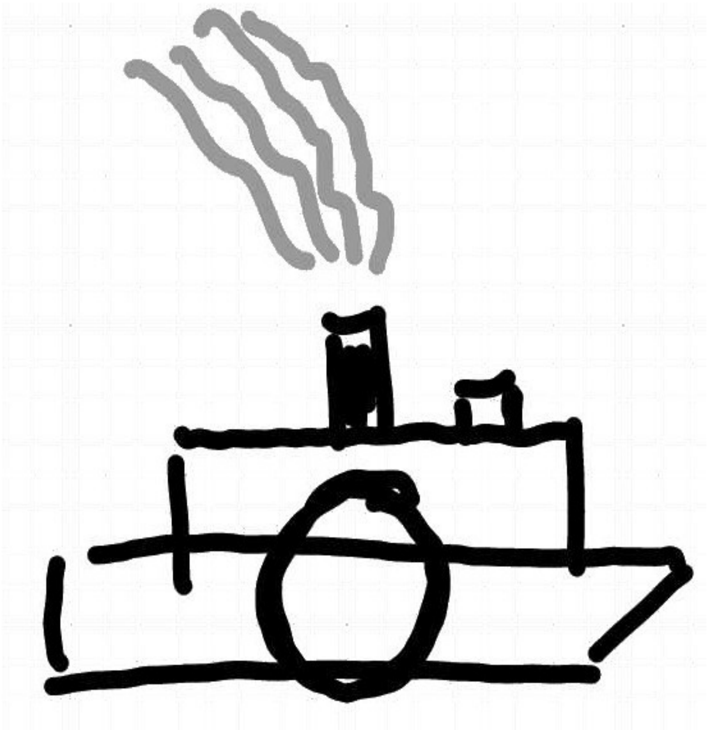
Track your deployments (what was when deployed)

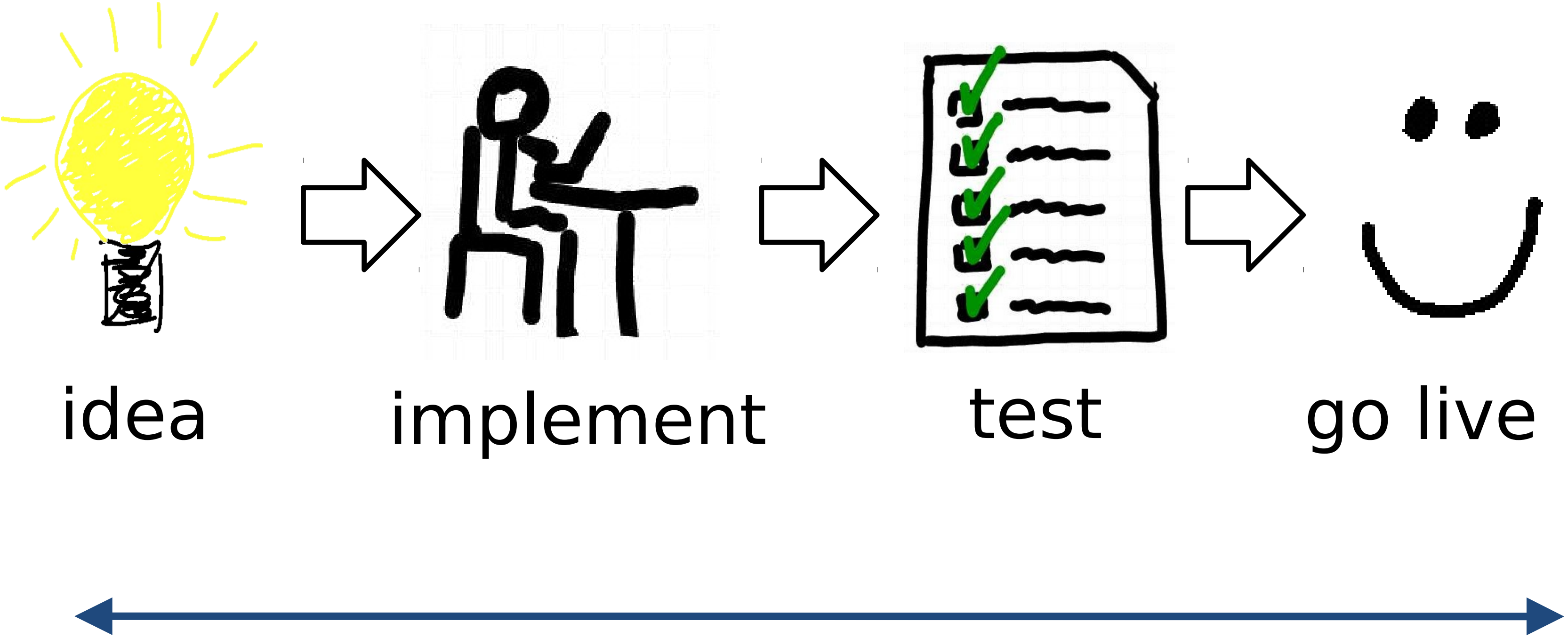
Lessons learned & good practices

Automate the configuration, too



Intro — Tests — Deployment — Build Pipeline — Docker

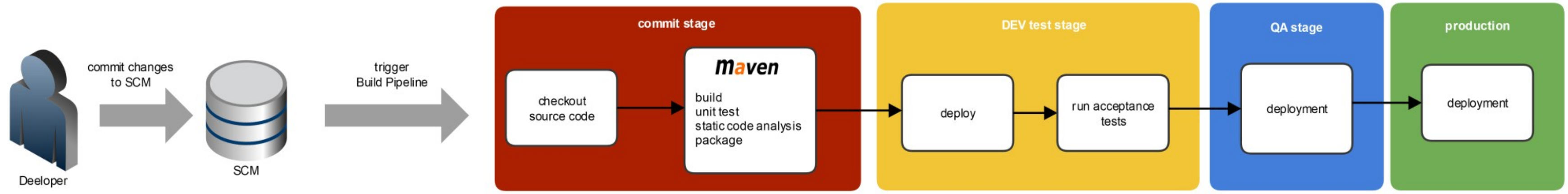




idea → implement → test → go live

cycle time

Intro — Tests — Deployment Build Pipeline — Docker



Deeloper

commit changes to SCM

SCM

trigger Build Pipeline

commit stage

checkout source code

maven

build
unit test
static code analysis
package

DEV test stage

deploy

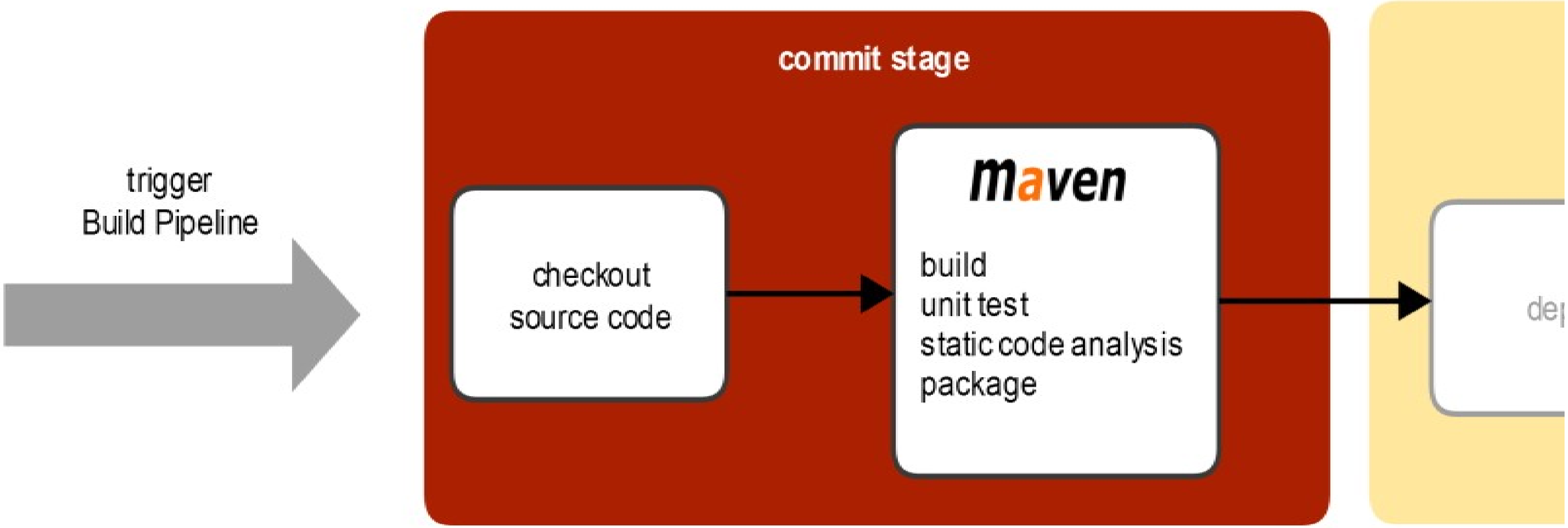
run acceptance tests

QA stage

deployment

production

deployment



trigger
Build Pipeline

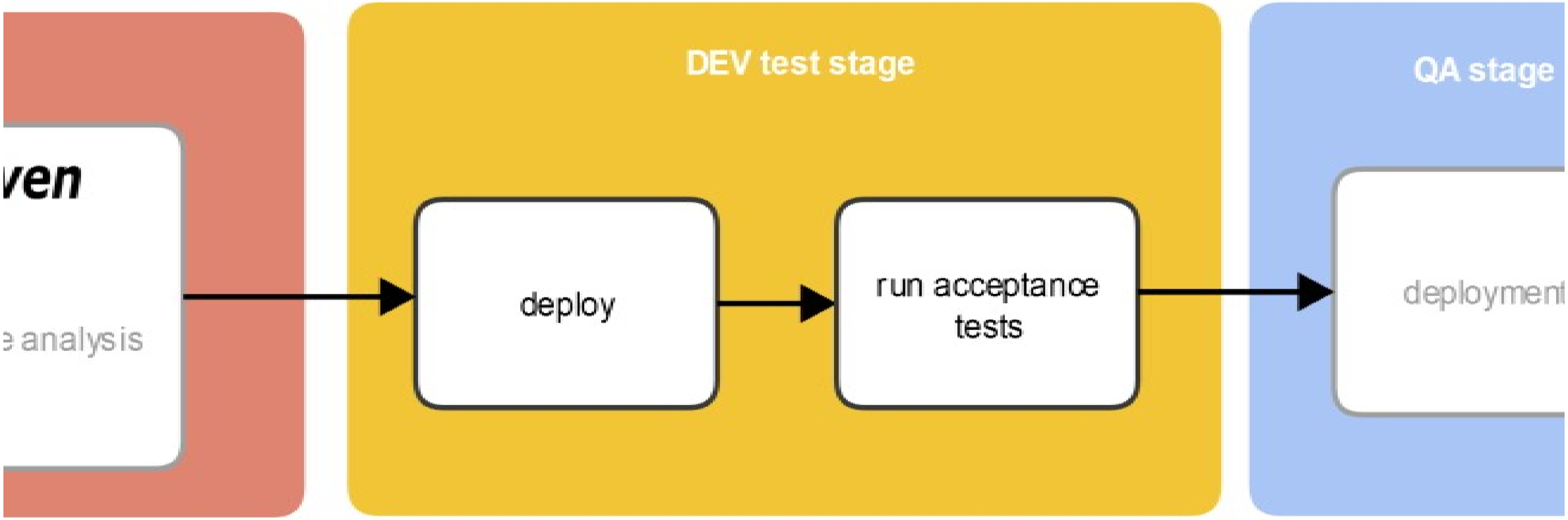
commit stage

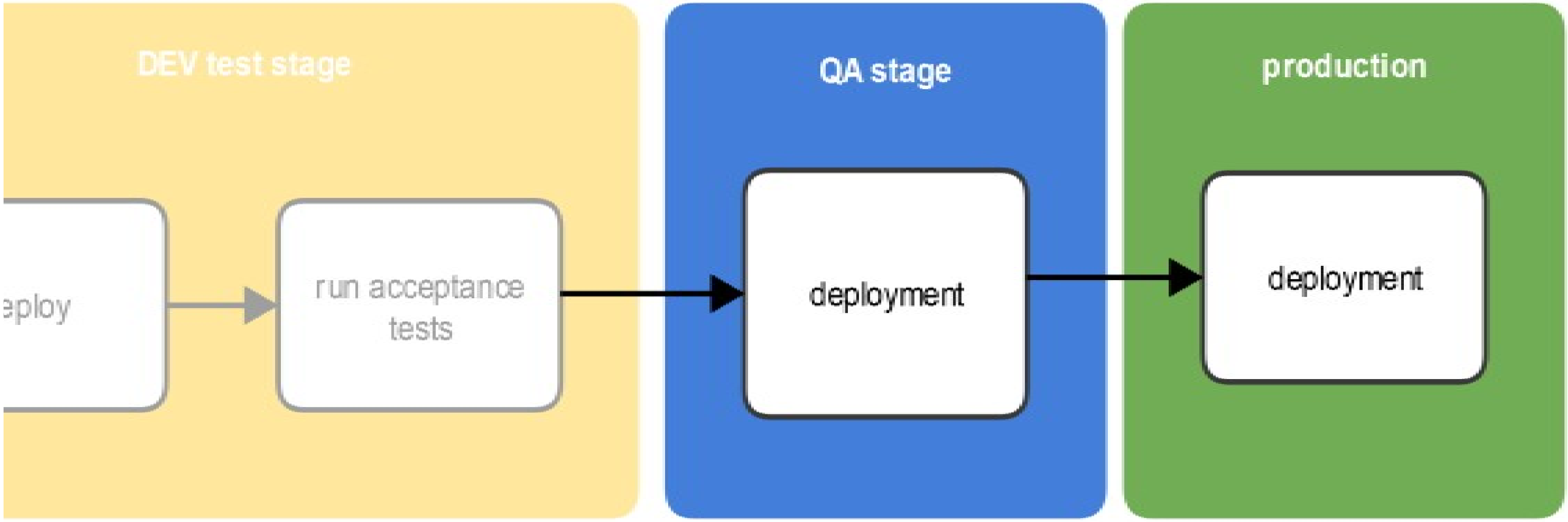
checkout
source code

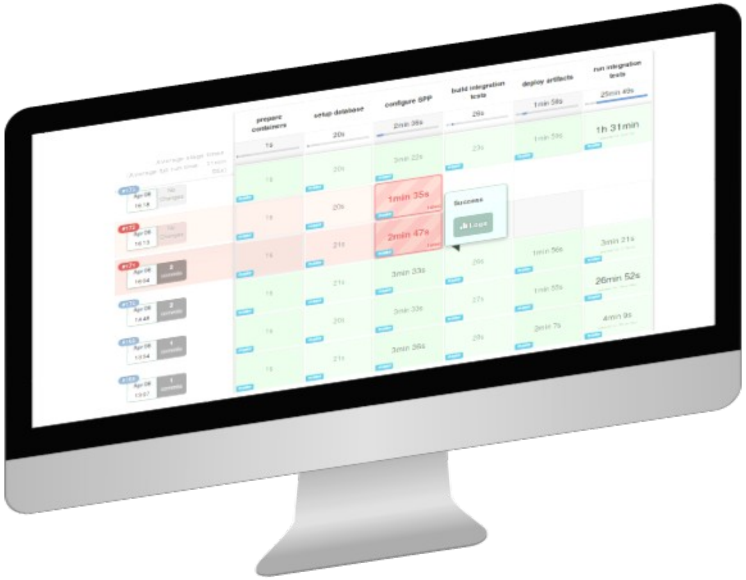
maven

build
unit test
static code analysis
package

dep

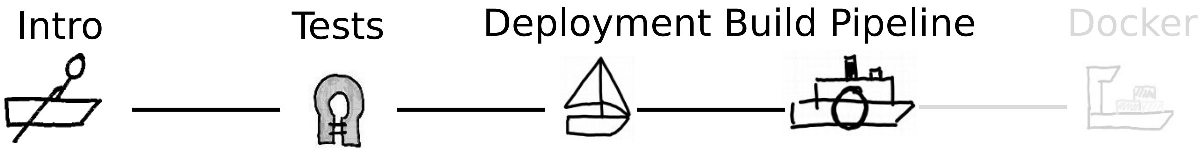

ven
e analysis
DEV test stage
deploy
run acceptance tests
QA stage
deployment


DEV test stage
eploy
run acceptance tests
QA stage
deployment
production
deployment

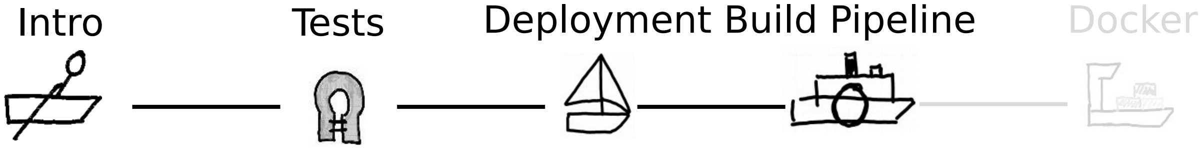


Jenkins pipeline plugin

- Groovy DSL
  - flexibel
  - pipeline as code
- supports stages
- Docker support



Intro Tests Deployment Build Pipeline Docker

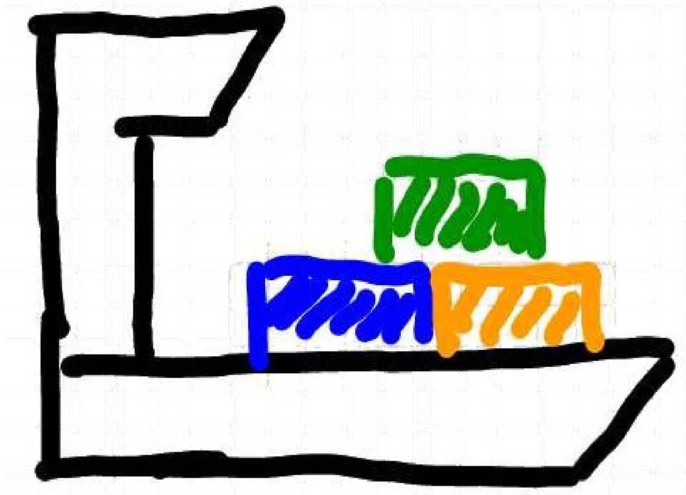

Intro

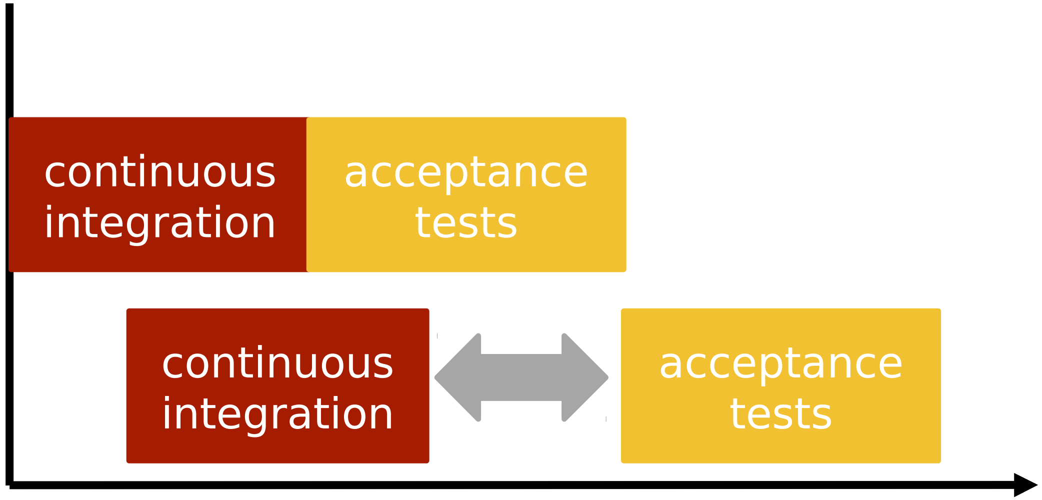
Tests

Deployment

Build Pipeline

Docker





continuous integration

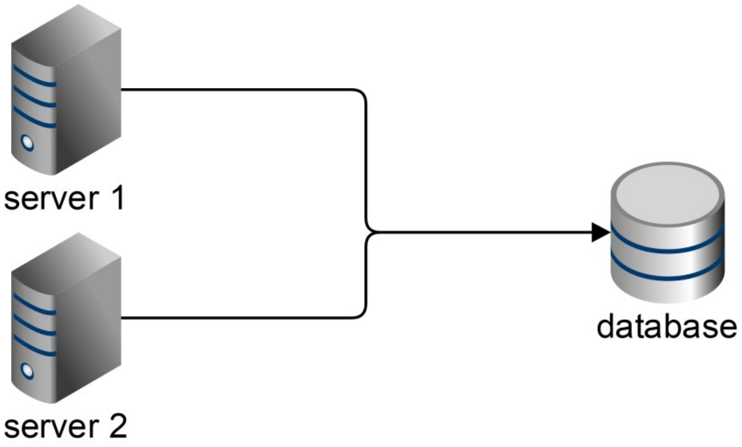
acceptance tests

continuous integration

acceptance tests

```
srv1 = docker
    .image('wildfly')
    .run("--name s1 “ +
            "--link db:db")
```

```
testImg
   .inside('--link s1:s1 ‘ +
              '--link s2:s2')
{
    // tests
}
```



```
srv2 = docker
    .image('wildfly')
    .run("--name s2 “ +
            "--link db:db")
```

```
dbContainer = docker

.image('postgres:9.4')
    .run("--name db")
```

always stop your containers

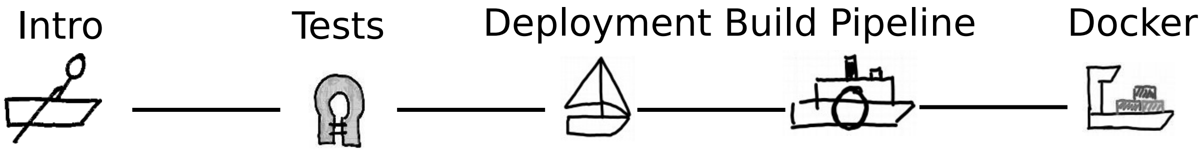
```
try {  /* ... */
} finally {
  con?.stop()
}
```

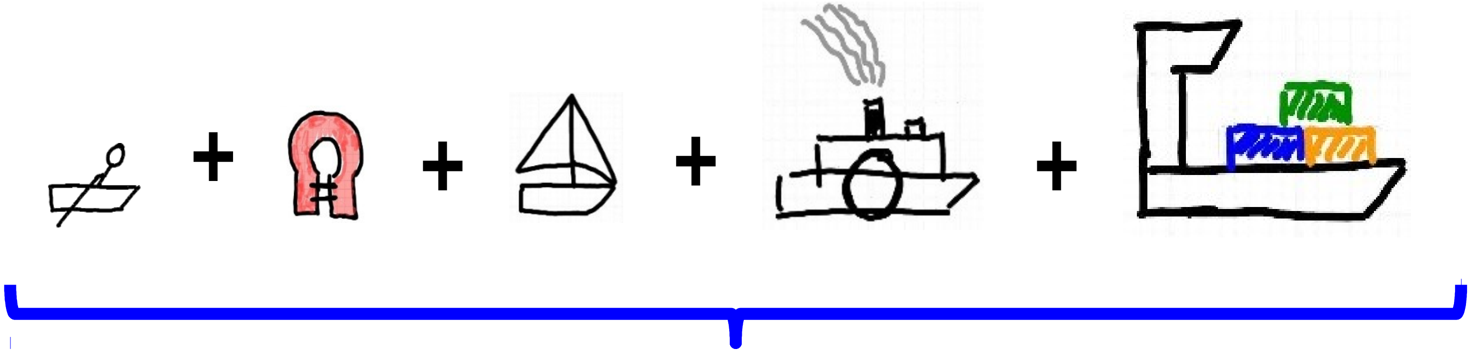
Save log files and data before stopping the container

Pipelines
using containers
lessons learned

wait until the container has started

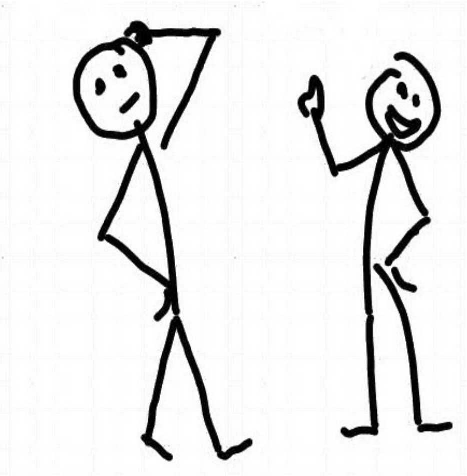
Use unique names for your containers or none





Continuous delivery

WE ARE
HIRING



Tobias Getrost
Lead Developer
1&1 Telecommunication
tobias.getrost@1und1.de