Deploy anytime

How we reduce deployment costs

Today’s deployment

- Manual activity from a amount of operation staff
- Planning needs to be done at least one weak beforehand because of the work council
- Deployment only happens in off-peak hours
- Downtime during deployment
- as this a manual task, it’s not reproducible and we had more often longer downtime as planned
- After code is in test we need some weeks until code is released with a lot of different changes
- Code testing is a manual step, therefore we cannot test every think every deployment.

Vision

- Deploy anytime
- No downtime
- No man day's for deployment
- Be in production after 2 hours, after code change
- All test (functional, performance, security) are done automatically
- Less error’s as no manual action are involved
- Rollback can be done through operation team, with no extra information
- Deployment in A/B Version to compare customer success

Step’s to go

- No Big Bang project
- Every week we’ll introduce a new small improvement out of the area:
  - Enhance Monitoring.
  - Enhance testing
  - Automate compile/application deployment
  - Automate Rollback
  - Introduce A/B Load Balancing (old/new Version)

Revenue

- No dowtime during deployment saves 1h downtime for monthly deployment. Lose 10k € every hour system is down.

  → increase business volume 120k€/year.

- Save of staff Cost for deployment

  → save 2k € every deployment

What we need

- Organizational change - One Team (development and operations need to work more together)
- 20% of work time for implementing the new way in the next year
- Start with openSource Tools, but we might need to add support or other tools which will cost extra – also depending on our cloud strategy. (e.g. Gitlab, datadog, aws costs)