ISH

Insurance Solutions House AG

Building Products with DevOps Philosophy

Niels Humbeck (Product Manager: Car Insurance Solutions)

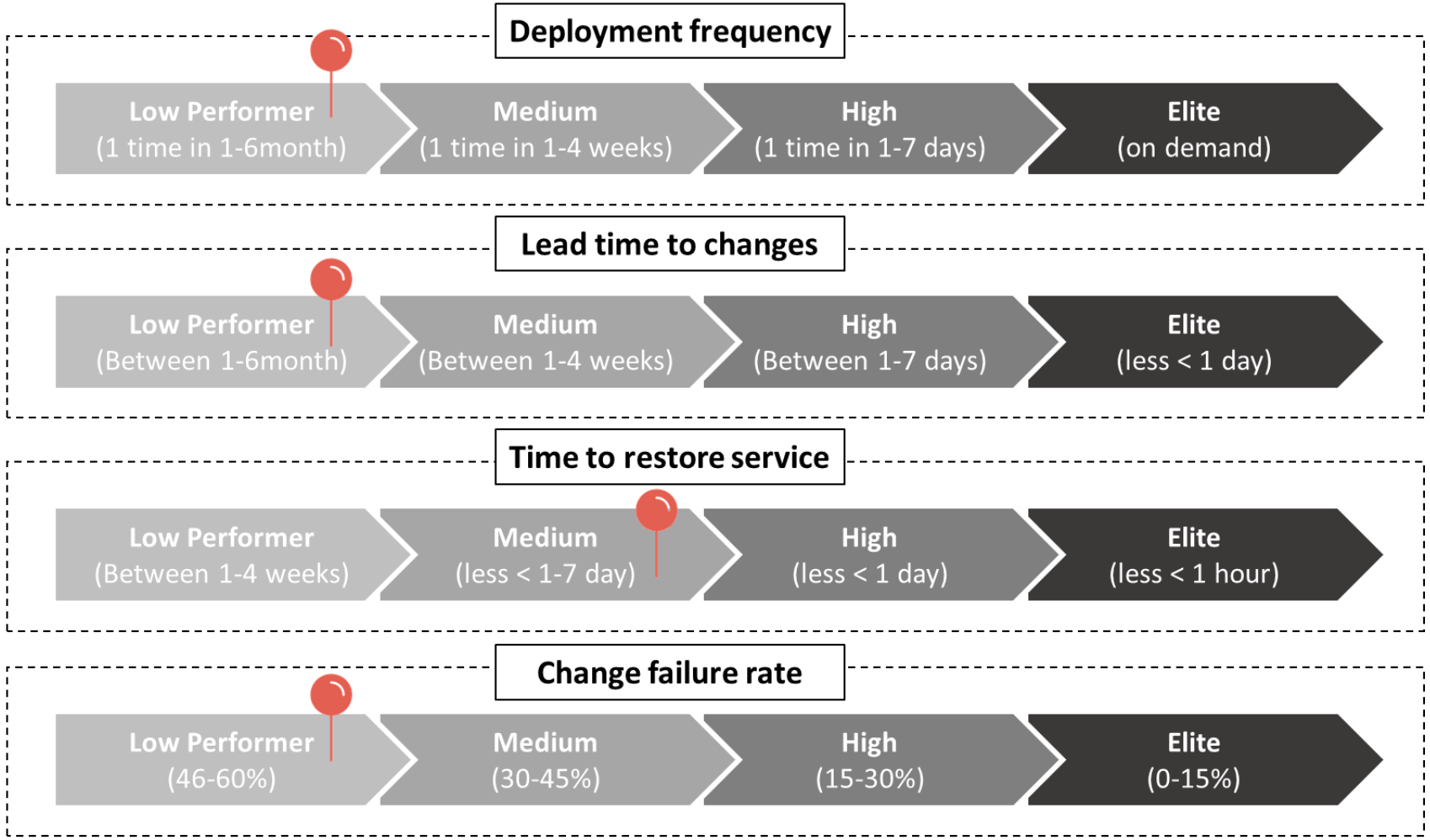
08.05.2021 Köln

Agenda

1. Benchmarking of our Software Lifecycle Management
2. What is DevOps?
3. Our next release: Mobile application for car insurance
4. Traditional Approach vs. DevOps Approach
5. Summary

Benchmarking our Software Lifecycle Management

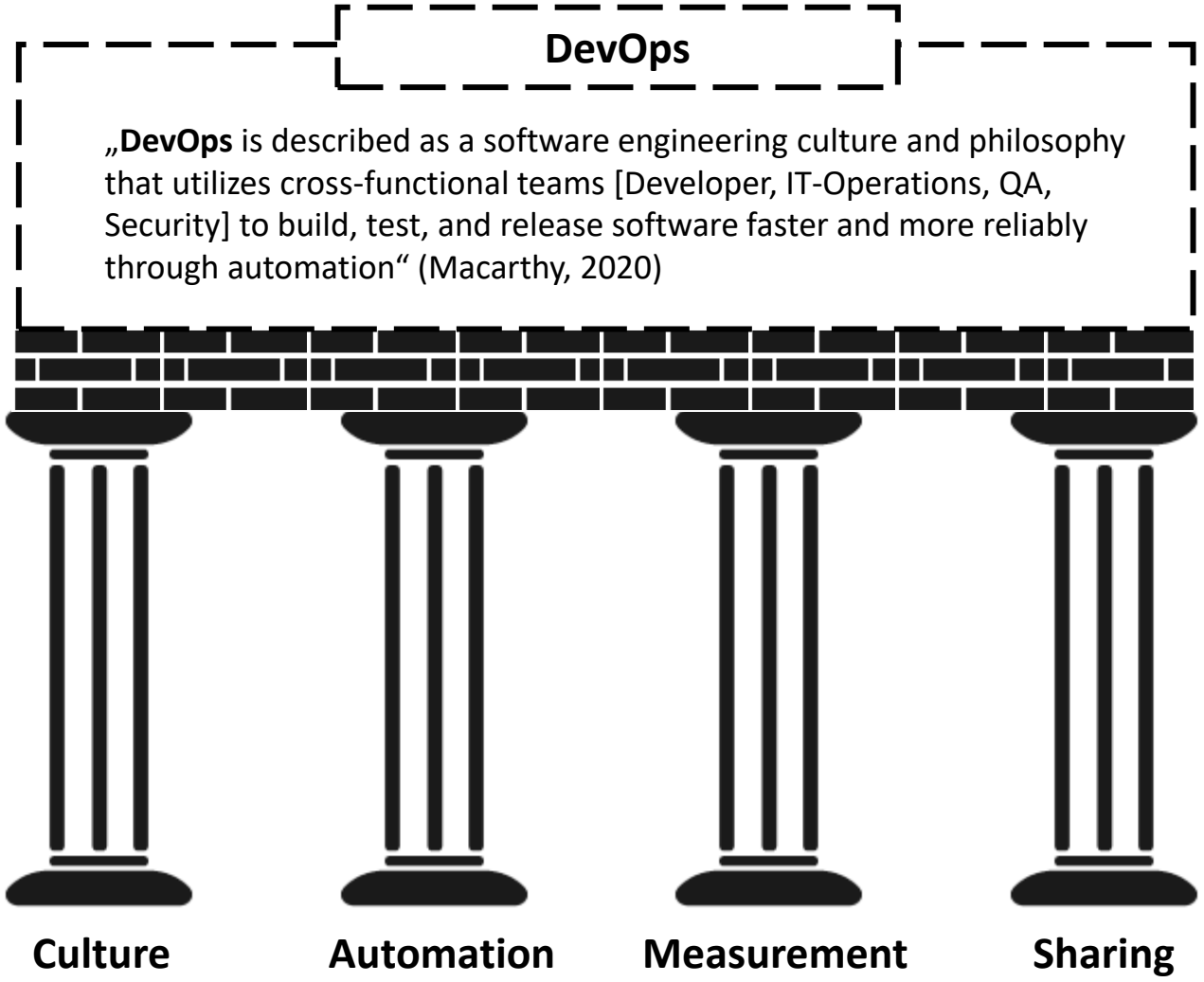
98% of the elite companies incorporate DevOps philosophy!

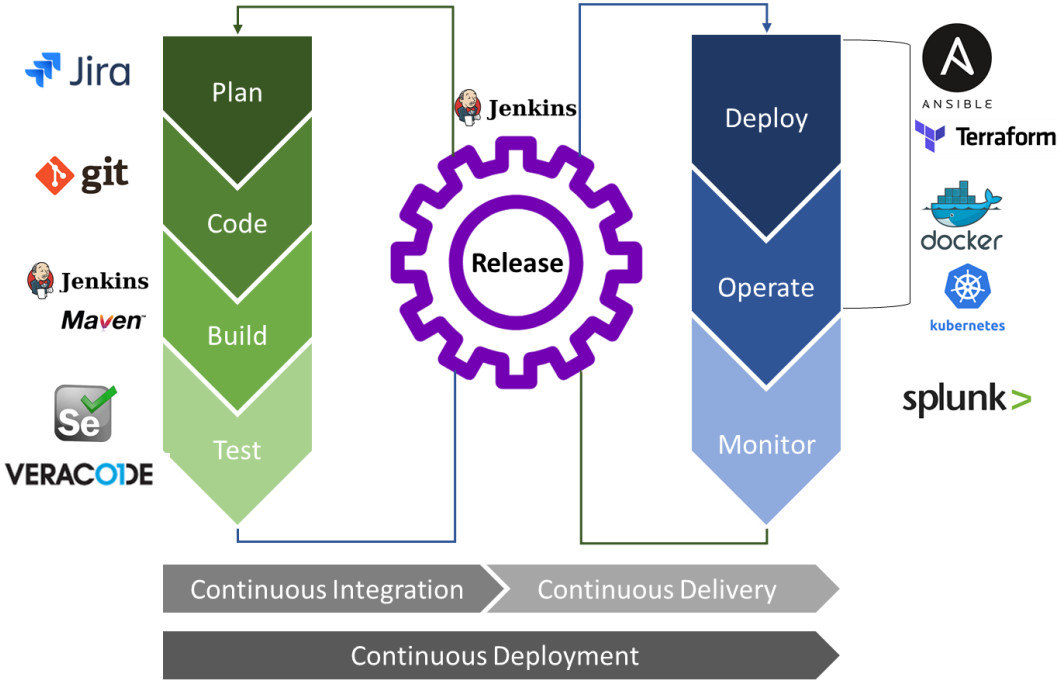


Our position

Portman(2020), Puppet(2015)

What is DevOps?





Macarthy (2020), Halstenberg (2020), Lwakatare (2016), Fernandez (2015) , Hüttermann (2012)

What is DevOps?

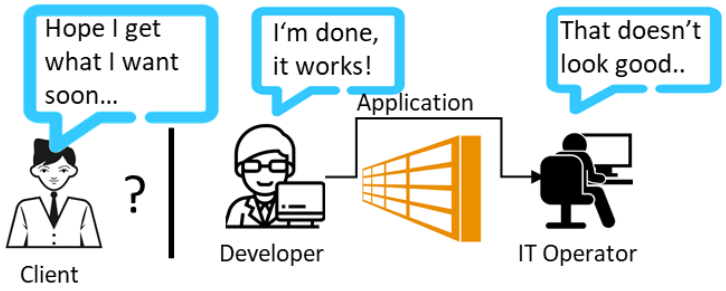
High level comparison: Traditional Software Lifecycle Management vs. DevOps

Traditional Approach

- ➢ Culture of collaboration
- Skill centric silo organization (functional organization)
- Extensive alignment & central planning is needed
- Conflicting objectives (developers: agility & innovation versus IT Operators: stability & optimization) leading in poor collaboration
- Business Stakeholder are poorly included in the software development process

- ➢ Degree of automatization
- Low automation of workflows and processes
- Manual Configuration of production environment
- Mainly manual testing

- ➢ Access to shared information between departments is limited

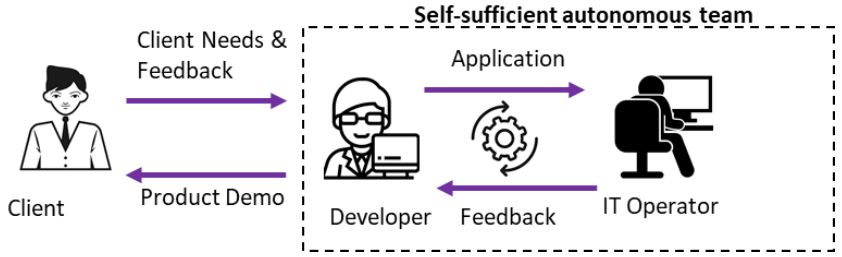


DevOps Approach

- ➢ Culture of collaboration
- Cross functional teams (streamlined organization)
- Releases are teamwork with shared commitment towards product
- DevOps as a culture to bridge different objectives of Devs & Ops
- Business Stakeholder are included in the software development process

- ➢ Degree of automatization
- High automation of workflows and processes
- Infrastructure as code: Automated provisioning, configuring & managing infrastructure
- Fully automated test

- ➢ Developers have real time access to troubleshooting information



Halstenberg (2020), Kapadia (2015), Lawakatare (2016), Farcic (2016), Fernandez (2015)

Our next release: Mobile application for car insurance

Staying ahead of our competitors

Market Position

- Europeans leading software developer for car insurance market
- Sales Teams in 16 Countries in EU
- Generates 25% our sales & 30% our earnings

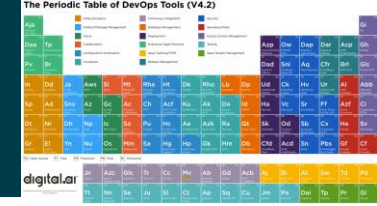
Market Research

- Increasing competition through FinTechs
- Market share decreases 2% (YtD)
- Customer complaints
  - Process for contracting takes to long
  - Too much paper-based steps
  - No smartphone App
  - UI not customer friendly
  - Time to market for new releases takes to long

User Requirements

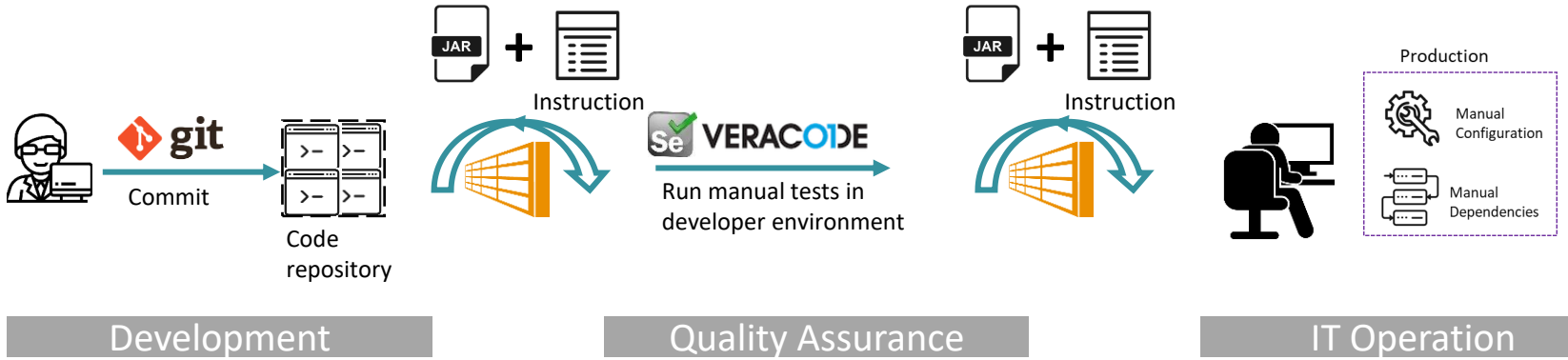
- Mobile Application
- Time to contracting < 30 min.
- Min. 2 small releases each month
- Customer friendly UI
- Go live in 6 month

Traditional vs. DevOps: Build and deliver process



Flow of code: Degree of integration & automation

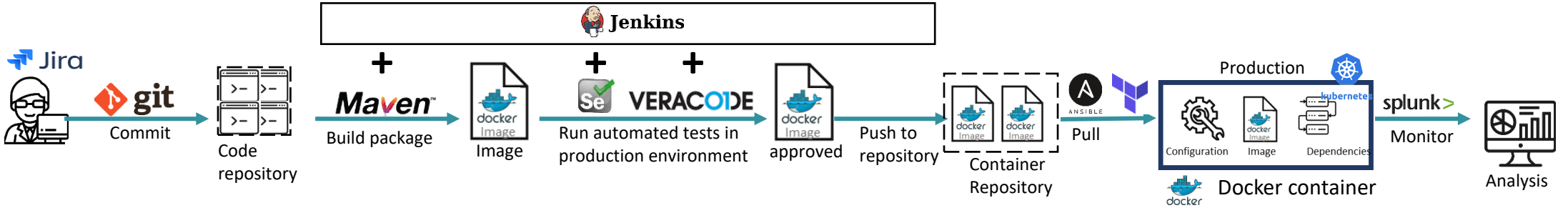
Traditional Approach



Characteristics:

- Discontinuous flow with process interruptions
- Huge amount of work in progress
- Low automation degree
- No feedback loops

DevOps Approach



Plan → Code → Build → Test → Release → Deploy → Operate → Monitor → Plan

CF → Continuous Integration (CI) → Continuous Delivery (CD) → Continuous Feedback (CF)

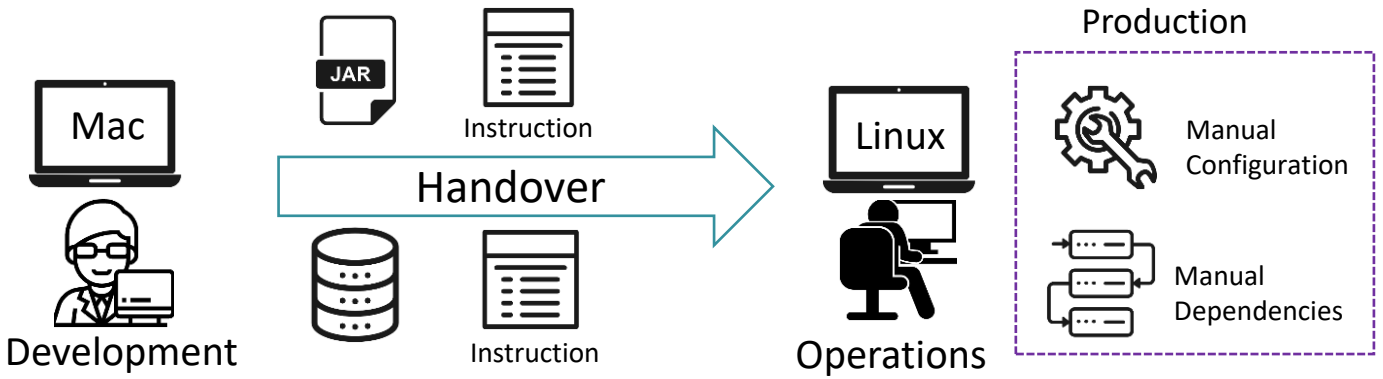
Continuous Deployment (CD)

Halstenberg (2020), Farcic (2016), TheDev(2019), Digital.ai (2020)

Traditional vs. DevOps: Build and deliver process

IT Infrastructure & Deployment process

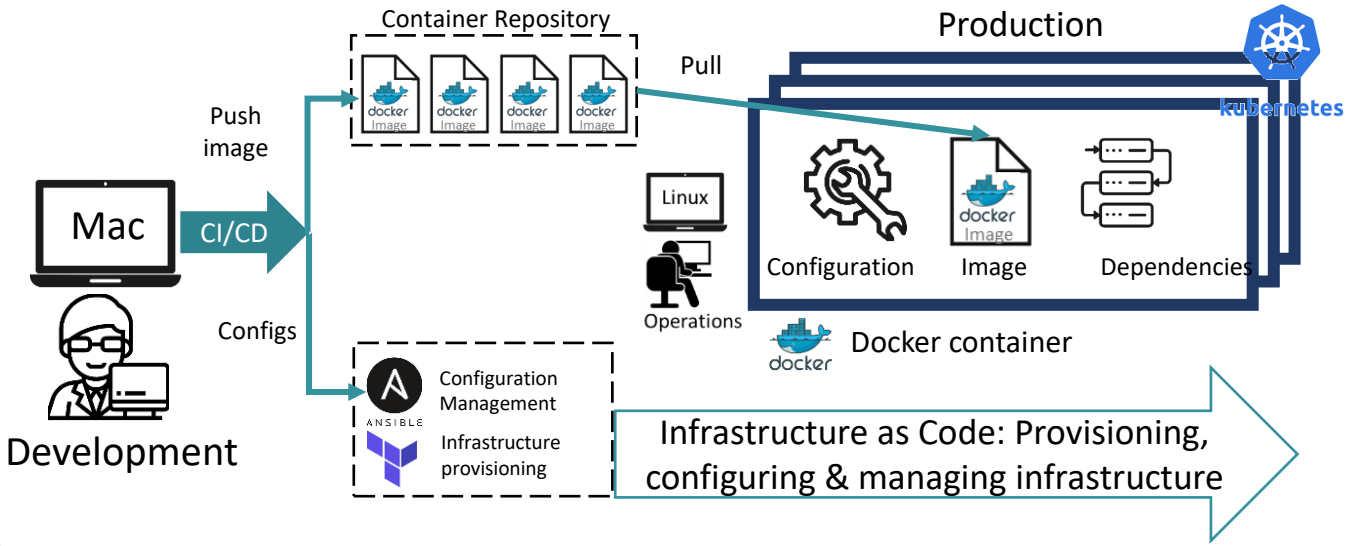
Traditional Approach



Summary:

- Manual provisioning, configuring & managing infrastructure
- Installation complexity of services is high
- Configuration & installation in life production
- Possible conflicts with dependencies are likely
- Instructions can be misunderstood

DevOps Approach



Summary

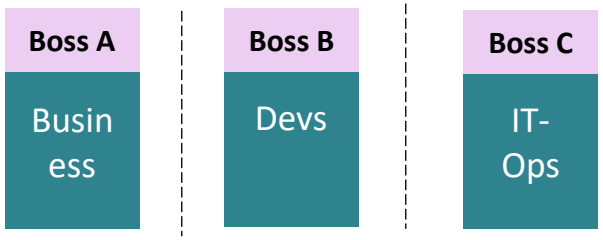
- Container: isolated environment, all necessary dependencies and configurations included, one command to install
- Infrastructure provisioning automated with Terraform
- Configuration management automated with Ansible
- Scalability with Kubernets
- Deployment of image with just one click (cont. delivery)
- Fully automated deployment is possible (cont. deployment)

Janashia(2020A-C), Farcic (2016), Pumperla(2020)

Traditional vs. DevOps: Team composition

Traditional Approach

Organization Structure



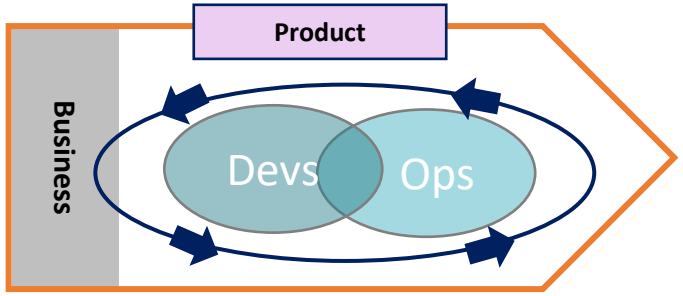
- Skill centric silo organization
- Commitment towards function

Team structure & Culture

- Commitment towards function not product
- Central scheduling
- No end-to-end product vision
- High-qualified engineers (experts in one area)
- Multi management (each silo has a leader)
- "Do not fail" culture
- Huge alignment effort needed
- Dismissive information sharing
- Definition of done: "I did my job"

DevOps Approach

Organization Structure



- Streamlined organization
- Cross functional teams: mixed personal (Devs & Ops)
- Commitment towards product

Team structure & Culture

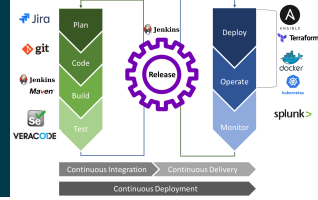
- Small teams working on a micro service (Amazons two pizza rule)
- Product ownership sharing
- End-to-end product vision
- Self-organization & autonomy
- High qualified engineers
- Skills over roles
- Single management (product owner is leader)
- Fail small, early and recover fast to drive innovations
- Strong collaboration
- Definition of done: "Readiness for deployment"

Change management

- Cultural change is severe
- Organizational changes needs strong change leadership
- Learning of new technologies is time consuming

Lwakatare (2016), Macarthy (2020), Halstenberg (2020), Fernandez (2015), Pumperla (2020), Skeleton(2013), Kapadia (2015)

Summary



	Traditional Approach	DevOps
Culture of Collaboration	• Skill centric silo organization • Commitment towards function	• Streamlined organization with cross functional teams • Commitment towards product
Automation	• Low degree of automation	• High degree of automation: CI/CD, Continuous Feedback, Infrastructure as Code, Automated testing
Measurement	• Focus: cost & capacity	• Focus: continuous flow of code, lead time to changes, deployment frequency, time to restore, change failure rate
Sharing	• Dismissive information sharing • Developers have no real time access to troubleshooting information	• Intensive sharing of near-realtime generated actionable information's

Benefits of DevOps

- Strengthen collaboration of Devs & Ops in one cross functional team (incl. security & testing)
- Shared commitment towards product
- Measurable KPIs:
  - Increased deployment frequency
  - Reduction of lead time to changes
  - Reduction of time to restore service
  - Reduction of change failure rate

Drawbacks of DevOps

- Cultural change is severe, need for change management
- Learning of new technologies is time consuming
- Reinvention of processes cost time & resources
- Productivity will decrease in transition phase

My suggestion: Lets go DevOps!

iubh

ISH

Q&A

Library (1/3)

- Arora, S. (2021), Best DevOps Tools to Learn and Master, https://www.simplilearn.com/tutorials/devops-tutorial/devops-tools, last access 19.04.2021 at 7pm
- Digital.ai (2020), The Periodic Table of DevOps Tools (V4.2); https://digital.ai/sites/default/files/pictures/2020-06/Digital.ai_Periodic-Table-of-DevOps.pdf, last access: 19.04.2021 at 7pm
- Farcic V. The DevOps 2.0 Toolkit. Packt Publishing; 2016. Accessed November 3, 2020. http://search.ebscohost.com.pxz.iubh.de:8080/login.aspx?direct=true&db=nlebk&AN=1345213&site=eds-live&scope=site
- Fernandez, D., Diaz, J., Garcia, J., Perez, J., Gonzalez-Prieto, A., (2015), DevOps Team Structures: Characterization and Implications
- Halstenberg, J., Pfitzinger B., Jestädt, T. (2020), DevOps – Ein Überblick, Springer Vieweg
- Hüttermann, M. (2012), DevOps for Developers, Apress, Berkeley, CA, https://doi-org.pxz.iubh.de:8443/10.1007/978-1-4302-4570-4
- Janashia, N. (2020A), Docker Tutorial for Beginners, https://www.youtube.com/watch?v=3c-iBn73dDE, last access: 01.05.2021 at 7pm
- Janashia, N. (2020B), Terraform explained in 15 mins, https://www.youtube.com/watch?v=l5k1ai_GBDE, last access: 1.05.2021 at 7pm
- Janashia, N. (2020C), DevOps Roadmap 2021 - How to become a DevOps Engineer?, https://www.youtube.com/watch?v=9pZ2xmsSDdo, last access: 1.05.2021 at 7pm
- Kapadia, M. (2015), Comparing DevOps to traditional IT: Eight key differences, https://devops.com/comparing-devops-traditional-eight-key-differences/, last access: 19.04.2021 at 7pm

Library (2/3)

- Lipnitski, A. (2019), DevOps Implementation: Where to Start and how to make it result in Success, https://www.scnsoft.com/blog/devops-implementation-guide#:~:text=%20DevOps%20implementation%20roadmap%20%201%20Organizing%20a,Docker%20solves%20the%20problem%20with%20the...%20More%20, last access: 19.04.2021 at 7pm
- Lwakatare, L., Kuvaja, P., Oivo, M., (2016), An exploratory study of DevOps - Extending the dimensions of DevOps with practice, Faculty of information and electrical engineering, university of Oulu Finland
- Macarthy RW, Bass JM. An Empirical Taxonomy of DevOps in Practice. 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Software Engineering and Advanced Applications (SEAA), 2020 46th Euromicro Conference on. August 2020:221-228. doi:10.1109/SEAA51224.2020.00046
- Portman, D. (2020). Are you an Elite DevOps performer? Find out with the four keys project; Google Cloud, https://cloud.google.com/blog/products/devops-sre/using-the-four-keys-to-measure-your-devops-performance, last access: 30.04.2021 at 13:13
- Prof. Dr. Pumperla, M (2020)., Software Engineering for Data Intensive Sciences (DLMDSSEDIS01), IU International Universit of Applied sciences)
- Puppet (2015), 2015 State of DevOps Report
- Puppet, Circleci (2020), State of DevOps Report 2020
- Robinson, M. (2020), Is it possible to transit to DevOps from Waterfall?, https://www.cprime.com/resources/blog/it-possible-to-transition-to-devops-from-waterfall/#:~:text=At%20their%20core%2C%20waterfall%20and,different%20teams%20to%20work%20collaboratively., last access: 19.04.2021 at 7pm

Library (3/3)

- Skeleton, M. (2013), What team structure is right for DevOps to flourish, https://blog.matthewskelton.net/2013/10/22/what-team-structure-is-right-for-devops-to-flourish/, last access 02.05.2021 at 10 am
- Söllner, D., Ingianni, L. (A 2017), Podcast – Folge 1: Was ist DevOps?; https://open.spotify.com/show/063ybYc7UyWJjjdFYXgPRe, last access 19.04.2021 at 7pm
- Söllner, D., Ingianni, L. (B 2017), Podcast- Folge 2: Das Periodensystem der DevOps-Tools; https://open.spotify.com/show/063ybYc7UyWJjjdFYXgPRe, last access 19.04.2021 at 7pm
- TheDev, J. (2019), The eight phases of a DevOps Pipeline, https://medium.com/taptuit/the-eight-phases-of-a-devops-pipeline-fda53ec9bba, last access: 01.05.2021 at 6:31 pm

Library Icon

- https://thenounproject.com/search/?q=speach+bubble&i=2315266 
- https://thenounproject.com/search/?q=organisation&i=2738585 
- https://thenounproject.com/search/?q=performance&i=3865086 
- https://thenounproject.com/term/wall/3022598/ 
- https://thenounproject.com/term/programmer/1277160/ 
- https://thenounproject.com/term/programmer/2767105/
- https://thenounproject.com/term/businessman/2498231/ 
- https://thenounproject.com/term/bridge/1233928/ 
- https://thenounproject.com/term/market-research/3456124/ 
- https://thenounproject.com/search/?q=Requirements&i=3430242 
- https://thenounproject.com/search/?q=position&i=2906205 
- https://thenounproject.com/term/thumbs-down/61041/ 
- https://thenounproject.com/term/thumbs-up/61040/ 
- Foto 1ste Seite: https://thenounproject.com/photo/pattern-cubes-4dEanb/

Library Icon

- https://thenounproject.com/search/?q=Laptop&i=2687479
- https://thenounproject.com/search/?q=jar&i=3404101
- https://thenounproject.com/search/?q=database&i=1378907
- https://thenounproject.com/search/?q=instructions&i=3536947
- https://thenounproject.com/term/data-dependency/3157942/
- https://thenounproject.com/term/configuration/1546936/
- https://thenounproject.com/search/?q=file&i=1558376
- https://thenounproject.com/term/analysis/349960/
- https://thenounproject.com/term/pillar/985572/
- https://thenounproject.com/search/?q=walls&i=1557679