DevOps in a nutshell

Dirk Lehmann, SAP
Disclaimer

The information in this presentation is confidential and proprietary to SAP and may not be disclosed without the permission of SAP. Except for your obligation to protect confidential information, this presentation is not subject to your license agreement or any other service or subscription agreement with SAP. SAP has no obligation to pursue any course of business outlined in this presentation or any related document, or to develop or release any functionality mentioned therein.

This presentation, or any related document and SAP's strategy and possible future developments, products and or platforms directions and functionality are all subject to change and may be changed by SAP at any time for any reason without notice. The information in this presentation is not a commitment, promise or legal obligation to deliver any material, code or functionality. This presentation is provided without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. This presentation is for informational purposes and may not be incorporated into a contract. SAP assumes no responsibility for errors or omissions in this presentation, except if such damages were caused by SAP’s intentional or gross negligence.

All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.
About me

- With SAP since 2001
- Implemented the first daily delivery process at SAP with twogo (2014)
- Current job: Operations at SAP CP ABAP Environment (a.k.a. Steampunk)
- Co-Organizer DevOpsDays Zurich https://devopsdays.ch
- Conference speaker https://doergn.github.io/
- @doergn
There is no Canonical Definition of DevOps

“a cross-disciplinary community of practice dedicated to the study of building, evolving and operating rapidly-changing resilient systems at scale.”

Jez Humble

“DevOps is... an umbrella concept that refers to anything that smoothes out the interaction between development and operations.”

Damon Edwards

“DevOps is a cultural and professional movement. The best way to describe devops is in terms of patterns and anti-patterns.”

Adam Jacobs

“the emerging professional movement that advocates a collaborative working relationship between Development and IT Operations, resulting in the fast flow of planned work (i.e., high deploy rates), while simultaneously increasing the reliability, stability, resilience and security of the production environment.”

Gene Kim
What’s the problem?

- DEV
- OPS
What’s the solution?

- DEV

- OPS
The term DevOps is **bad. Really bad. Like really really bad!**

- It started as “Agile Operations”, turned into DevOps and at the time everyone realized it was a terrible mistake the term was already established.

### What about...

- **Security?**  
  - *DevSecOps?*
- **Business?**  
  - *DevBizOps?*
- **Quality?**  
  - *DevQOps?*
- **UA?**  
  - *DevUAOps?*
- ...

### Don’t try to create a new bad term. Let’s live with one bad term: DevOps
Based on a blog entry by John Willis: http://itrevolution.com/devops-culture-part-1/
Based on a blog entry by John Willis: http://itrevolution.com/devops-culture-part-1/
Based on a blog entry by John Willis: [http://itrevolution.com/devops-culture-part-1/](http://itrevolution.com/devops-culture-part-1/)
Based on a blog entry by John Willis: [http://itrevolution.com/devops-culture-part-1/](http://itrevolution.com/devops-culture-part-1/)
<table>
<thead>
<tr>
<th>Role</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>PM</td>
<td>1</td>
</tr>
<tr>
<td>Dev</td>
<td>1</td>
</tr>
<tr>
<td>QA</td>
<td>1</td>
</tr>
<tr>
<td>UA</td>
<td>1</td>
</tr>
</tbody>
</table>

Based on a blog entry by John Willis: [http://itrevolution.com/devops-culture-part-1/](http://itrevolution.com/devops-culture-part-1/)
Based on a blog entry by John Willis: http://itrevolution.com/devops-culture-part-1/
Based on a blog entry by John Willis: [http://itrevolution.com/devops-culture-part-1/](http://itrevolution.com/devops-culture-part-1/)
Based on a blog entry by John Willis: http://itrevolution.com/devops-culture-part-1/
Based on a blog entry by John Willis: http://itrevolution.com/devops-culture-part-1/
Banana Corp.

Lead time:
25 days

Based on a blog entry by John Willis: http://itrevolution.com/devops-culture-part-1/
Based on a blog entry by John Willis: http://itrevolution.com/devops-culture-part-1/

Banana Corp.

Lead time:
25 days
Lead time:
25 days

Based on a blog entry by John Willis: [http://itrevolution.com/devops-culture-part-1/](http://itrevolution.com/devops-culture-part-1/)
Banana Corp.

Lead time:
22.5 days

Based on a blog entry by John Willis: [http://itrevolution.com/devops-culture-part-1/](http://itrevolution.com/devops-culture-part-1/)
Tear Down Silos

• **Biggest** problem in most organizations is **not in the silos**

• **Biggest** problem in most organizations is **between the silos**

• Create **cross-functional, autonomous teams**
  - All people with a direct impact on the product should work in one team
  - Keep teams small -> Trust is the foundation of DevOps

• **Functional silos only make sense in not direct product related tasks**
  - E.g. HR, Legal, ...
In Other Words…

“DevOps: A culture where people, regardless of title or background, work together to imagine, develop, deploy and operate a system.”

Ken Mugrage, Thoughtworks
CALMS

Culture
Automation
Lean
Measurement
Sharing
From: Functional Silos  Cross-functional teams
From: Heteronomy  Autonomy
From: Avoid Risk and Failure  Failing as Learning Opportunity
From: Never Revise Policy  Continuous Process Review
CALMS

Culture

Automation

Lean

Measurement

Sharing

- Infrastructure-as-Code
- Configuration-as-Code
- Everything-as-Code
- Testing
- Creates Consistency
- Avoid Repetitive Tasks
- Fast
- Cattles over Pets
- Shares and documents Knowledge
CALMS

Culture

• System thinking

Automation

• Always focus on the customer

• Fail early, fail often

Lean

• Eliminate Waste

• Continuous Improvement

Measurement

Sharing
CALMS

Culture

Automation

Lean

Measurement

Sharing

- “You can only improve what you can measure”
- Builds Confidence
- Identifies Bottlenecks
- Measure and use result as input for next improvement cycle
- Hypothesis Driven Development
CALMS

Culture

- Share Resources (a.k.a. Cloud Computing)
- Share Responsibility
- Share Success
- Share Knowledge

Automation

- Within Team
- Between Teams
- Between Companies

Lean

Sharing
Agile Organization Setup
Agile Organization Setup
Conway’s Law

“Organizations which design systems [...] are constrained to produce designs which are copies of the communication structures of these organizations.”

- Architectures are constrained to organizations communication structures
- In doubt, communication structures overrule architecture...

“Culture eats strategy for breakfast”

Peter Drucker
Conway’s Law - Reversed
From Organization Setup…

- Setup your organization
From Organization Setup…to microservice architecture…

- Setup your organization in alignment to your architecture
From Organization Setup...to microservice architecture...to containers

• Setup your organization in alignment to your architecture and leverage containers as fast, reproducible, encapsulated runtimes
From Lighthouse projects to Cloud Core DNA
Deploy with Confidence (DwC)
Culture, Process, Tools, Real Life Example

October 2020

Storage Location
Deliver Mesh of 3 Microservices
Traditional Way at SAP (Multi-Branch, Deployment = Release)

200 Developer Organization, transformed to Cloud Engineering in 9 month, by...

Branches: Master **Preprod** Release FIX Hotfix

Dev Commit

**To Preprod**
(Prepare a release, test phase, …)

Issue found in PreProd test fix needed

Release 2005
(Test was finally okay, release to customer)

Challenge 1: Complexity in Branch and Patch management
- You will only see when you are live! Then it is too late

Challenge 2: Explosion of combinations of services and branch-versions that integrate.
Have these combos been tested???
Below just example with 2 services, 3 Branches

Often > 14 Weeks between releases

Challenge 4: How often has the big release integration caused a big bang at customer? How long did it take for Dev team to solve issue that have been developed months back?

Challenge 5: How long does it take to follow the exceptional hotfix process that is usually not tested?

Challenge 6: Stress and peak load for our engineers.

Deploy with Confidence
Process and Culture

Branches: Master

Service 1

Service 2

Service 3

28,000 Increments in CALM: Usually behind Toggle

Feature 1

Feature 2

Feature 3

Service 2

Service 1

Feature 1

Feature 2

Branches:

**Master**

**Preprod**

**Release**

**FIX**

**Hotfix**

Deploy to PROD with Confidence

Yesterday #110
Activation (Release)
Feature 1
Feature 2
Feature 3

_today 24 hours later_

Today #111 09:00 CET
Activation (Release)
Feature 1
Feature 2
Feature 3

Tomorrow #112

Hotfix Deploy
just 30 Minutes including all compliance and tests

Challenge 3: How frequent can deliveries be done here?
Considering the copies, rebuilds, test phases.

Challenge 5: How long does it take to follow the exceptional hotfix process that is usually not tested?

Challenge 6: How often has the big release integration caused a big bang at customer? How long did it take for Dev team to solve issue that have been developed months back?

200 Developer Organization, transformed to Cloud Engineering in 9 month, by…
Cloud Engineering **Process** - Deploy with Confidence
Delivering SaaS Apps at Scale, Daily

- **Customer Requirements**
- **Plan**
- **Code**
- **Build**
- **Test**
- **Deploy**
- **Release**
- **Operate, Observe and automate**

**Define feature** including DoDs feature needs to fulfill before it gets released

**Development** behind release-feature **toggle**

**Single Trunk**

**Fast component build pipeline** with lead time <15 min. Compliancy!

**Fully automated end to end tests** for every single feature from customer perspective

**Deployment** != Release

**Release = activate feature**
strict fulfillment of DoDs

**Zero downtime activation**
without interrupting existing user sessions

**Progressively**

**Measure feature usage/success**

© 2020 SAP SE or an SAP affiliate company. All rights reserved. | PUBLIC
Cloud Engineering **Tools** - Deploy with Confidence
Golden Path of Tools to deliver SaaS Apps at Scale, Daily

**Legend:**
- Golden Path
- Free Breakout
- Open Source/Commercial off the shelf
- SAP proprietary tools

**Customer Requirements**
- Plan
- Code
- Build
- Test
- Deploy
- Release
- Operate, Observe and automate

**Tools**
- Jira
- CAP
- Jenkins
- GitHub
- Spring Boot
- Node.js
- SauceLabs
- Piper
- webdriverIO
- Nightwatch.js
- Azure DevOps
- MTA
- CF PUSH
- SlackBot
- MS-Teams
- Docker
- ELK
- Dynatrace
- Sentry
- HANA
- Postgres
- WebAssist

Often several tools can be used to live the process. Tools have individual pro/cons. People have choice here and are enabled to optimize for all (inner-source)
### Comparing Software Delivery Performance - DORA

<table>
<thead>
<tr>
<th>Aspect of Software Delivery Performance*</th>
<th>Elite</th>
<th>High</th>
<th>Medium</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Deployment frequency</strong>&lt;br&gt;For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?</td>
<td>On-demand (multiple deploys per day)</td>
<td>Between once per day and once per week</td>
<td>Between once per week and once per month</td>
<td>Between once per month and once every six months</td>
</tr>
<tr>
<td><strong>Lead time for changes</strong>&lt;br&gt;For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?</td>
<td>Less than one day</td>
<td>Between one day and one week</td>
<td>Between one week and one month</td>
<td>Between one month and six months</td>
</tr>
<tr>
<td><strong>Time to restore service</strong>&lt;br&gt;For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?</td>
<td>Less than one hour</td>
<td>Less than one day*</td>
<td>Less than one day*</td>
<td>Between one week and one month</td>
</tr>
<tr>
<td><strong>Change failure rate</strong>&lt;br&gt;For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?</td>
<td>0-15%&lt;sup&gt;h,c&lt;/sup&gt;</td>
<td>0-15%&lt;sup&gt;h,d&lt;/sup&gt;</td>
<td>0-15%&lt;sup&gt;h,d&lt;/sup&gt;</td>
<td>46-60%</td>
</tr>
</tbody>
</table>

* The KPIs define how to measure DevOps capabilities and have been defined in a large study called DORA State of DevOps.
Key Take-Aways

Deploy with Confidence (DwC) is a SAP-Internal, proven combination of
• Culture
• Processes
• Tools
 for daily delivery to production. – With Confidence!

SAP Cloud ALM, Live.
From monolithic development to daily delivery!
200 Engineers, 9 months,
✓ We deliver Daily.
✓ We improve Daily Automation, Co-innovations, Inner-Source, …
✓ We get faster every day!

SAP ALM Summit 2020 – Topic 2020-10-09

<table>
<thead>
<tr>
<th>Time</th>
<th>Track 3 - Cloud Technologies for ALM professionals</th>
</tr>
</thead>
<tbody>
<tr>
<td>09:30</td>
<td>Introduction to cloud technologies</td>
</tr>
<tr>
<td>10:15</td>
<td></td>
</tr>
<tr>
<td>10:25</td>
<td>Workflow with SAP Cloud Platform</td>
</tr>
<tr>
<td>11:10</td>
<td></td>
</tr>
<tr>
<td>11:20</td>
<td>Introduction to extensions for SAP Cloud Platform with Cloud Foundry, Cloud SDK and SAP API Business Hub</td>
</tr>
</tbody>
</table>
DevOps Community Topic Page

- One entry page for DevOps with SAP
- Launched Sep/2019
- Community for and by SAP, Customers and Partners
DevOps with SAP

Vision

“Apply **DevOps** principles to customer’s **SAP technology stack** (and beyond).

Along the **CALMS** pattern, we’ll provide tools, libraries, best-practices, trainings, services and knowledge exchange.”
DevOps in a Nutshell

October 8, 2020
3 PM – 4:30 PM, CET (Germany)

Speakers:
Dirk Lehmann,
DevOps evangelist
and Continuous
Delivery expert, SAP

Andreas Löbel,
Chief
Development
Architect for SAP
Cloud ALM, SAP

DevOps in an ABAP-based environment

October 15, 2020
3PM - 4:30 PM, CET (Germany)

Speaker:
Karin Spiegel,
Product Management team member for Cloud and Lifecycle Management, SAP SE

Efficient DevOps with SAP
Cloud Platform

• October 22, 2020
• 3PM - 4:30 PM, CET (Germany)

• Speaker:
  • Boris Zarske,
  • Product Manager for Cloud & Lifecycle area, SAP SE
Thank you.