



assertEquals(SCM, DevOps)=True?

Lars Bendix
bendix@snescm.org

*Scandinavian Network of Excellence
in Software Configuration
Management
(sneSCM.org)*

<http://cs.lth.se/~bendix/Research/SCMnDevOps/>



Acknowledgements

Erik Hochbergs
erik.hochbergs.175@student.lu.se
Christian Pendleton
cpe@pragma.net

*Praqma AB
Malmö
Sweden*

Laroy Nilsson Sjö Dahl
fpr08lni@student.lu.se

*Department of Computer Science
Lund University
Sweden*



Agenda

Agenda:

- You: What is DevOps?
- You: Why DevOps?
- LBx: Why DevOps?
- LBx: What is DevOps
- SCM in a slide or two
- assertEquals(SCM, DevOps) = True?
- the infinite loop
- pizza & beer & mingling – & escape ;-)
- detailed/in-depth discussions – Q&A



You – DevOps what?

Bright ideas from a brilliant crowd:



You – DevOps why?

Bright ideas from a brilliant crowd:



Why DevOps?

Spotify:

- they don't know their customers
- they can't talk to their customers
- they can **monitor** what their customers **do**
- they want to bring **ideas** into production fast
- **if** something catches the customers' attention:
 - it is further developed/explored
- **if not:**
 - it dies
- it is a "requirements engineering" method
- with emphasis on "**fast from idea to production**"
- **fast from question to answer**



What is DevOps?

Culture (People, processes and tools)

Automation (Do it once, do it twice – automate)

Lean (Continuous improvement and learning)

Measurement (development, production, business)

Sharing (Collaborate, give feedback, don't copy)

What more is DevOps than "just": Dev + Ops?

How does DevOps differ from Agile/Scrum/XP – or CD?

Is it "monitoring and measuring users" to know what they like?



DevOps characteristics

| | | |
|------------------------|------------------------|-----------------------|
| Automation | | Sharing |
| | Continuous Integration | |
| Feedback loops | Shift left | Continuous Testing |
| | Continuous Delivery | |
| Infrastructure as Code | Culture | Continuous Deployment |
| Monitor | Change management | Collaboration |
| Planning | Agile & Lean | Quality Assurance |

from [Hochbergs, Sjödaahl, 2020]



What is SCM?

SCM is Software Configuration Management

Part Configuration Management:

- Identification (configuration items, traceability)
- Control (change process, change requests, CCB)
- Status Accounting (recording and reporting information)
- Audit (assesses compliance with requirements before acceptance of change into a baseline)

Part Software:

- Build management
- Process management (adherence to development processes)
- Environment management
- Teamwork (facilitate team interactions)
- Defect tracking



What is SCM?

- Coordination problems: SD, DM, SU
- Coordination strategies: locking, copy/merge, split/combine, Long Transactions (LT) / Strict LT (SLT)
- branching strategies
- history and diffing
- CI, CMDB, traceability, BoM
- CSA
- CR, change process, CBB
- CA, baselines

SCM is method/paradigm agnostic



Why SCM?

What is the purpose/goals of SCM:

Sometimes it is embarrassing to be a computer programmer. What other profession has such a remarkable rate of schedule and cost overrun and outright failure? [...]

Our failures are not of the individual contributors; most of us design, code and debug adequately or even well. Rather, the failure is one of coordination. Somehow we lack the ability to take 20 or 30 good programmers and meld them into a consistently productive team.

Wayne Babich, 1986



Why SCM?

What is the purpose/goals of SCM:

I use the term in a more expansive sense. I include not just the formal release of software to the customer, but the day-to-day and minute-by-minute evolution of the software inside the development team. Controlled evolution means that you not only understand *what* you have *when* you are **delivering it**, but you also understand *what* you have *while* you are **developing it**.

Wayne Babich, 1986



assertEquals(SCM, DevOps)=True?

Claims:

- same goals: "efficient quality"
- service and support a development organization
- create a lean and mean delivery machine



SCM = DevOps?

SCM is: "an infrastructure where things and information can move around in a structured, controlled and smooth way to come together in a defined product"

DevOps is: "the ability for organizations to efficiently deliver services and applications"



SCM = DevOps?

DevOps is: "a set of practices intended to reduce the time between committing a change to a system and the change being placed into normal production, while ensuring high quality"

SCM is: "this cool stuff that will facilitate a team to coordinate people and things so they can carry out changes in an orderly fashion right from idea/conception to production - and retirement - and avoid any chaos and confusion in the process"

Ops: resist change

Dev: live on change – often and fast

SCM can make both happy – plus more (PM, company, customer)

Can DevOps be done @ Airbus?



assertEquals(SCM, DevOps)=True?

Both are focused on:

- establishing development infrastructure
- organizing developers' environments
- quality assurance (quality gates)
- infrastructure management
- release management



assertEquals(SCM, DevOps)=True?

Claims:

- DevOps is about making software developers fully aware of and *accountable for* SCM
- Focused on continuous integration and continuous delivery
- DevOps can be viewed as Lean applied to SCM



assertEquals(SCM, DevOps)=True?

Observations:

- DevOps is a term used to describe the coordination between Development and Operations
- SCM is a *means* of achieving this coordination



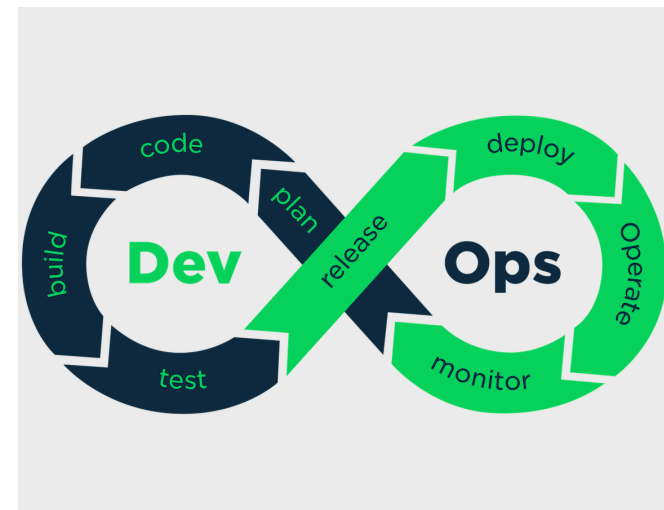
assertEquals(SCM, DevOps)!=True?

Differences:

- CM is not a methodology
- purpose of CM is not user feedback – or...
- a CM is **not** a programmer, tester, QA, PM, ...
- CM is providing "infrastructure" for those people to collaborate



assertEquals(SCM, DevOps)=True?





Discussion – Q&A

Brad Appleton (and Mario Moreira):

- assertEquals(**agile**SCM,DevOps)=True
- Agile SCM is the deadbeat dad of DevOps

Carsten Poulsen:

- assertEquals(SCM,DevOps)!=True
- DevOps is a production automation process
- SCM is an archive process



Discussion – Q&A

SCM in a DevOps context:

- what are the things you "care about"?
- why are you using git flow or PR?
- should the project have a clear goal?
- what are the questions you frequently ask?
- do you love other people's sloppy code?
- why are you using feature branches?
- is this Done – DoneDone – ReallyDone – ...?