

Building a Collaborative Culture

A Grounded Theory of Well Succeeded DevOps Adoption in Practice

Welder Pinheiro Luz ¹ Gustavo Pinto ² Rodrigo Bonifácio ³

¹ Brazilian Court of Accounts

² Faculty of Computing, Federal University of Pará

³ Computer Science Department, University of Brasília

October 12, 2018

research context

Brazilian Federal Court of Accounts (TCU)

Characteristics

- ▶ a huge number of enterprise systems
- ▶ prevalence of JEE architecture using a shared domain model

Brazilian Federal Court of Accounts (TCU)

Characteristics

- ▶ a huge number of enterprise systems
- ▶ prevalence of JEE architecture using a shared domain model
- ▶ clear separation between development and production teams
- ▶ rigid time frames for publishing software assets (once a week)

well known problems

well known problems



Use of agile development practices

well known problems



Use of agile development practices, though expecting delays when publishing the software into acceptance testing and production environments.

well known problems

well known problems



The program works at the development environment, but it does not at the acceptance testing and production environments.

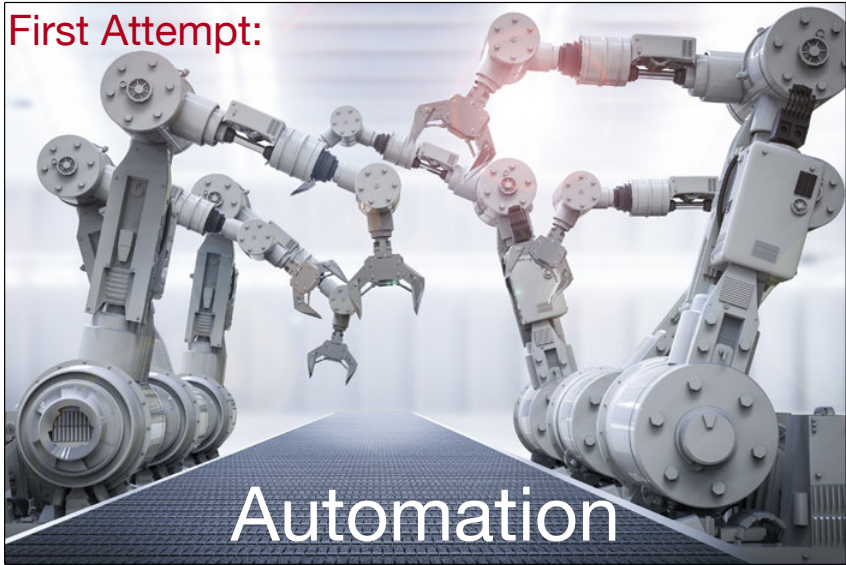
well known problems



The program works at the development environment, but it does not at the acceptance testing and production environments. “It must be a bug of the program, so the development team must take care of it”.

Let's try fixing this issue using a **DevOps** approach!!!

First Attempt:



Automation





research

goal: understand and characterize DevOps

method: (classical) grounded theory approach

research question: what are the recommendations for DevOps adoption?

Grounded Theory

(1)

Open Coding Data
Collection

(2)

Selective Coding Data
Analysis

(3)

Theoretical Coding

Action Research

(Application of the Theory - 12 people involved)

Focus Group

(Validation - 4 participants)

Results

Core Category

Core Category

Collaborative Culture

Core Category

Collaborative Culture

- ▶ Removing silos between development and production teams

Core Category

Collaborative Culture

- ▶ Removing silos between development and production teams
- ▶ Concepts:
 - ▶ software development empowerment
 - ▶ operations tasks should be performed by development teams

Core Category

Collaborative Culture

- ▶ Removing silos between development and production teams
- ▶ Concepts:
 - ▶ software development empowerment
 - ▶ operations tasks should be performed by development teams
 - ▶ product thinking
 - ▶ straightforward communication

Core Category

Collaborative Culture

- ▶ Removing silos between development and production teams
- ▶ Concepts:
 - ▶ software development empowerment
 - ▶ operations tasks should be performed by development teams
 - ▶ product thinking
 - ▶ straightforward communication
 - ▶ blameless and shared responsibilities

Not only automation

*“During a process for DevOps adoption, there is a very strong cultural issue that the teams sometimes are not adapted to. Regarding that, **one thing that bothers me** a lot and that I see very often is **people hitching DevOps exclusively by tooling or automation.**”*

Product Thinking

*“We wanted to **hire people who could have a product vision**. People who could see the problem and think of the best solution to it, **not only thinking of a software solution, but also the moment when that application will be published**. We also brought together developers to reinforce that everyone has to think of the product and not only in their code or in their infrastructure.”*

Software Development Empowerment

*“It was not feasible to have so many **developers** generating artifacts and **stopping their work to wait for another completely separate team to publish it**. Or needing a test environment and having to wait for the operations team to provide it only when possible. These activities have to be available to quickly serve the development team.”*

Additional Categories

Additional Categories

Enabler categories

- ▶ automation
- ▶ transparency and sharing

Additional Categories

Enabler categories

- ▶ automation
- ▶ transparency and sharing

Categories of expected outcomes

- ▶ agility
- ▶ resilience (auto-scaling and recovering)

- ▶ two categories in a gray area: quality assurance and continuous measurement.

Collaborative Culture and Automation

When a developer needed to build a new application, the **previous workflow demanded him to create a ticket to the operations teams**, which should then manually evaluate and solve the requested issue.

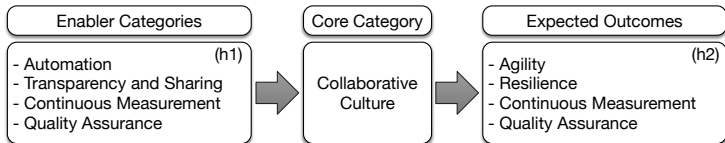
This task could take a lot of time and there was no visibility between teams about what was going on. . . . Today, **those silos do not exist anymore within the company**, in particular because **it is not necessary to execute all these tasks manually**.

Knowledge Sharing

So, here we have adopted this type of strategy that is the **infrastructure as code**, consequently we have the versioning of our entire infrastructure **in a common language**, in such a way that **any person**, a developer, an architect, the operations guy, or even the manager, **can understand the settings of an application** in a particular environment. So, transparency aggregates too much value for us . . .

A Theory for DevOps Adoption

A Theory for DevOps Adoption



Application at TCU

Application at TCU

- ▶ disseminate the needs for a collaborative culture
- ▶ promote tech talks and direct communication
- ▶ automation and (still limited monitoring) of several tasks
- ▶ infrastructure configuration as code

| System | Max. Number of Successful Builds in a Day |
|-----------------|---|
| e-TCE | 29 builds |
| e-Denuncia | 31 builds |
| e-Representacao | 33 builds |
| Conecta TCU | 42 builds |
| ... | |

| System | Max. Number of Successful Builds in a Day |
|-----------------|---|
| e-TCE | 29 builds |
| e-Denuncia | 31 builds |
| e-Representacao | 33 builds |
| Conecta TCU | 42 builds |
| ... | |

A general understanding that our model is fostering the adoption of DevOps practices at TCU.

Challenges

Challenges

- ▶ lack of a general understanding about DevOps
- ▶ core ideas do not reflect the structure of TCU
- ▶ limited use of monitoring services
- ▶ security concerns (developers do not understand security)

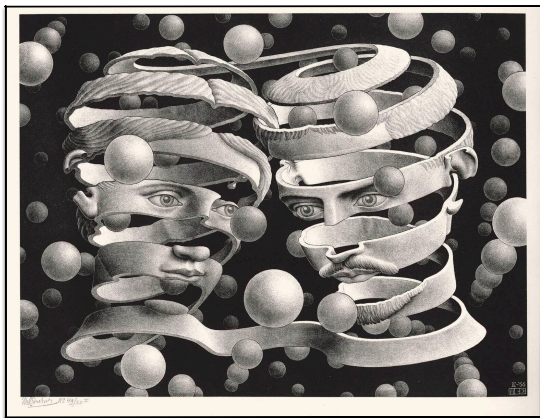
Personal Considerations

Personal Considerations

- ▶ Does DevOps mean the death of operation teams?

Personal Considerations

- ▶ Does DevOps mean the death of operation teams?
- ▶ Does DevOps practices introduce too much complexity?



Building a Collaborative Culture

A Grounded Theory of Well Succeeded DevOps

Adoption in Practice

Welder Pinheiro Luz ¹ Gustavo Pinto ² Rodrigo Bonifácio ³

¹ Brazilian Court of Accounts

² Faculty of Computing, Federal University of Pará

³ Computer Science Department, University of Brasília

October 12, 2018