



**HAL**  
open science

## Job Satisfaction Factors and Risks Perception: An embedded case study of DevOps and Agile Teams

Aymeric Hemon-Hildgen, Laetitia Monnier-Senicourt, Frantz Rowe

### ► To cite this version:

Aymeric Hemon-Hildgen, Laetitia Monnier-Senicourt, Frantz Rowe. Job Satisfaction Factors and Risks Perception: An embedded case study of DevOps and Agile Teams. Bridging the Internet of people, Data and things, ICIS International Conference on Information Systems, Dec 2018, San Francisco (CA), United States. hal-04576887

**HAL Id: hal-04576887**

**<https://hal.science/hal-04576887>**

Submitted on 15 May 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# **Job Satisfaction Factors and Risks Perception: An embedded case study of DevOps and Agile Teams**

*Completed Research Paper*

**Aymeric Hemon**

University of Nantes  
Chemin de la Censive du Tertre, 44322  
Nantes, France  
aymeric.hemon@univ-nantes.fr

**Laetitia Monnier-Senicourt**

University of Nantes  
Chemin de la Censive du Tertre, 44322  
Nantes, France  
laetitia.monnier-senicourt@univ-nantes.fr

**Frantz Rowe**

University of Nantes  
Chemin de la Censive du Tertre, 44322 Nantes, France  
frantz.rowe@univ-nantes.fr

## **Abstract**

*While Agile can be seen as an accelerator to develop software, it suffers from bottlenecks between development and operations. DevOps overcome agile bottlenecks providing better, quicker and safer deliveries through continuous delivery and deployment, improving end-to-end processes. However, literature on DevOps mechanisms and impacts is still scarce. Based on an exploratory case study within 11 agile and DevOps teams, this paper investigates first the perceptions of risks and difficulties when using DevOps and the difference with agile, second the effect of DevOps on job (dis) satisfaction. Findings shows DevOps as an enhancer of job satisfaction in comparison to agile. However, along with this impact this study unveiled a simultaneous effect of amplification of the risk with DevOps in comparison to Agile.*

**Keywords:** Job Satisfaction, Risks, Agile, DevOps

## **Introduction**

Since its inception the Information Systems (IS) function has been organized around a strict division of labor between Development and Operations functions. During the 1990's agile software development methods have been created and progressively adopted in order to deliver better, faster and safer applications (Humble and Farley 2010). This had considerable impact on the Development function itself and far lesser impact on the Operations function. Launched at the end of the 2000's, the DevOps movement can be seen as a systematic collaboration with operations members being part of development teams and associated with application development from the beginning of the development process. DevOps aims at enabling better, quicker and safer deliveries through continuous delivery and deployment improving end-to-end processes overcoming agile bottlenecks (Humble and Farley 2010; Kim et al. 2016). Consequently, DevOps as well as agile methods are spreading more and more within organizations (Dybå and Dingsøyr 2008; Version One 2018).

Whereas the scientific literature on agile is very rich both on agile adoption (Korhonen 2009; Sureshchandra and Shrinivasavadhani 2008), related satisfaction and performance (Tripp et al. 2016), the literature on DevOps phenomena is still scarce and DevOps not adequately studied (Erich et al. 2014; Smeds et al. 2015). While one study tried to find some perceived impediments to DevOps adoption (Smeds et al. 2015) Lwakatare, Karvonen, et al. (2016) identified some obstacles when moving towards DevOps in the Embedded Systems Domain. Nybom et al. (2016) studied the impact of mixing responsibilities between Dev and Ops and they found mixed results presenting DevOps approach benefits such as collaboration as well as new source for friction or risk (risk for holistically sub-optimal service configurations). Other studies investigated problems and challenges when adopting Continuous Delivery (CD) (Chen 2017; Laukkanen et al. 2017) which is only one of the multiple DevOps maturity level. To the best of our knowledge, very few scientific studies analyzed transition from agile to DevOps (Lwakatare, Karvonen, et al. 2016; Shropshire et al. 2017; Wiedemann 2017) as well as the risks and difficulties organizations face when moving to DevOps (Lwakatare, Karvonen, et al. 2016; Nybom et al. 2016). In particular knowledge on which difficulties and risks employees perceive in DevOps teams and how and why their (dis) satisfaction increases with DevOps in comparison with agile teams is lacking. This knowledge is important to succeed in the overhaul of the IS function which may come with DevOps, but also to make more realistic expectations about the value added of DevOps. Consequently, the goal of our research is to explore perceptions of employees regarding risks and difficulties and their job (dis)satisfaction and examine the extent to which they differ from the agile methods practice. The study of the perception of risks and difficulties crossed with the analysis of the sources of satisfaction and dissatisfactions could enlighten our understanding of the impact of DevOps use on job satisfaction and thus on performance as determined by Hackman and Oldham (1980).

Hence, our two research questions: Q1: What are the perceptions of job satisfaction, risks and difficulties of current software industry employees when using DevOps? Q2: Are there any differences in the perceived job satisfaction, risks and difficulties between Agile and DevOps approaches?

Answering these questions could help organizations to adapt and to move forward, keeping a high level of job satisfaction thus performance, to be more responsive and efficient avoiding pitfalls when moving to DevOps. The rest of the paper is structured as follow. In section 2, we discuss literature on Agility and DevOps and we present the Job Design framework. In section 3, we present the methodology followed for this exploratory research. In section 4 we present and illustrate our findings. Finally, in section 5, we discuss extensively our results regarding the literature and outline the validity of this research.

## **DevOps breaks silos between development and operations**

Agile software development can be traced back to the 1990's with the emergence of rapid application development (RAD) in 1991, later unified process (UP) and dynamic systems development method (DSDM), then Scrum in 1995, the most used agile method (Version One 2018). All these methods are considered as alternatives to traditional plan driven methods. The main goal of these agile methods is to deal with unfixed issues generated by traditional methods such as failure to comply with budget and deadlines, as well as little visibility on project progress creating a tunnel-effect until the final product delivery, or as the lack of responsiveness to changing or new requirements (Boehm and Turner 2005; Highsmith and Cockburn 2001). In 2001, the Agile Manifesto was published by 17 agile development

experts where they formulated 4 essential values and 12 principles for agile software development (Beck et al. 2001). Those 4 central values refer to the following: Individuals and interactions over processes and tools; Working software over comprehensive documentation; Customer collaboration over contract negotiation; Responding to change over following a plan. The latter points mean that agile methods have been designed to develop products in a changing environment, integrating evolutions regarding customer or user requirements through an iterative process with frequent deliveries. These agile methods imply new collaborations, e.g. between developers and users, and a higher level of interactions within the Think and the Build teams. Agile methods like Scrum or XP are mainly used by developers, therefore in the “Build” area in relationship with the “Think” area where Architects and Product Owners come from. However, with agile development challenges regarding flows, organization readiness, management and functional silos still exist, notably with operations (Boehm and Turner 2005; Dennehy and Conboy 2017; Nerur et al. 2005).

Hence, the move towards DevOps, launched in 2008, which responds to the enduring need to break silos within the IS function whose role is to think, build and run applications. DevOps could be considered as an extension of agility, like leading it to a higher level, but also different with larger scope and greater effects including speed of delivery (Bird, 2017). DevOps allows to perceive more opportunities and related risks in a larger environment but gives also more possibilities to make mistakes. What is clear is that without DevOps, agility remains in a paradigm with two separated functions. The studies and development area (Think and Build) follow logics of creativity and speed, whereas the operations area (Run) pursue logics of quality, stability and traceability. Agility is not resolving this problem of antagonistic logics because agility means the use of methods and practices within the Build area without breaking silos between Build and Run. DevOps is frequently presented as philosophy, or a movement, with related principles (Erich et al. 2014; Kim et al. 2016). Humble and Molesky (2011) highlighted four core values for DevOps: Culture, Automation, Measurement and Sharing. In their literature review on DevOps, Erich et al. (2014a) offered a precise description of these four pillars: culture is the extent to which an organization practicing DevOps attempt to eliminate cultural barriers between Dev and Ops areas; Automation is related to the high degree of automated processes adopted by the organization; Measurement refers to joint use of metrics used by both Dev and Ops, and to avoid using separate metrics; Sharing refers to the extent to which information is shared between Dev and Ops. These fundamental DevOps pillars are supported by principles mostly derived from Lean Thinking and Lean Management (Fitzgerald and Stol 2017; Humble et al. 2015; Kim et al. 2016). Kim et al. (2016, p.84) highlighted for example the principle of continuous learning across Dev and Ops teams characterized by “a high-trust culture that enables all departments to work together effectively, where all work is transparently prioritized and there is sufficient slack in the system to allow high-priority work to be completed quickly”. While barriers to knowledge sharing are an important issue in agile teams (Ghobadi and Mathiassen 2016), sharing is an essential element for collaboration across different specialties like Dev and Ops (Riungu-Kalliosaari et al. 2016). Moreover, whereas many DevOps practices are coming from agile practices, DevOps also elaborated a set of idiosyncratic practices directly linked to its four pillars, e.g. deployment process automation or Useful Metrics shared by Dev and Ops (Lwakatare, Kuvaja, et al. 2016).

## **From Agile and DevOps Practice to Job Satisfaction / Dissatisfaction**

### ***Perceived Risk to Agile and DevOps Practice***

Perceived risk is frequently associated to the acceptance and the practice of a product, service, technology or method. Risks related to agile acceptance have been investigated and partially evaluated (Boehm and Turner 2005; Elbanna and Sarker 2016). Other studies focused on risk management (Albadarneh et al. 2015; Kontio et al. 2004), risk linked to processes (Nyfjord and Kajko-Mattsson 2007) and practices like knowledge sharing (Ghobadi and Mathiassen 2016). Begel and Nagappan (2007) pinpointed perceived problems with agile, i.e. scaling up the project, frequency of meetings, losing sight of big picture. However, Nerur et al. (2005) emphasized that more caution is required when using agile methodology or incorporating them into existing practices. DevOps practices may face the same challenge as agile practices with risk. Erich et al. (2017) investigated DevOps adoption within six organizations and they found that all organizations were positive about their experiences, even if some of them encountered minor problems while implementing DevOps such as technical problems due to software architecture.

Wiedemann (2017) studied collaborations within an IT team on the move to DevOps and found that social capital could form key capabilities. Hemon et al. (2018) explored soft skills, competence and collaborations among teams and proposed a maturity model from agile to DevOps where they found that DevOps carries out greater smartness for the Information System function. Other authors worked on benefits and challenges regarding DevOps practice and they identified some inhibiting factors to DevOps practice, e.g. human factors, the lack of communication among employees, a resistance to change, and technical factors like technologies enabling complexity of development and deployment to production environments (Riungu-Kalliosaari et al. 2016).

### ***Agile and DevOps Practice as Enabler of Job Satisfaction / Dissatisfaction***

Various studies highlighted new ways of working, collaborating and investigated their effect on job satisfaction (Acuña et al. 2009; Tripp et al. 2016) within agile teams (Tessem and Maurer 2007) as well as comparing agile and non-agile team (Melnik and Maurer 2006). Tripp et al. (2016) demonstrated the direct effects of agile practice use on job satisfaction. Melnik and Maurer (2006) demonstrated that there are twice as many agile employees who are satisfied with their work as there are regular (waterfall) employees. Acuña et al. (2009) found that job satisfaction of agile team members is directly related to personality factors, i.e. agreeableness and conscientiousness. However, except on the last point, there is neither such literature on DevOps and job satisfaction, nor comparisons between DevOps and Agile teams. Some researchers investigated the impact of DevOps practice on workforce morale and they found that the fruits of DevOps implementation are not enjoyed. They explain it with a decline of organizational morale and more specifically they suppose that not all of the changes related to DevOps are viewed as positive (Shropshire and Sweeney 2017). Shropshire et al. (2017) tried to identify whether there is a personality more conducive to the practice of DevOps as Acuña et al. (2009) investigated for agile practice. They found two types of personalities, the first one supporting and enabling DevOps, the second one stifling innovation and slowing DevOps practice: “DevOps puts new demands on the organization. Some employees concentrate on the positives: improved software releases, increased uptime, improved collaboration, higher quality code, fewer errors, etc. Others are overwhelmed by its negative aspects: harder work, greater responsibility, blurred lines of authority, complex performance metrics, potential redundancies/layoffs, etc.” (Shropshire et al. 2017, p.1). These findings show how DevOps affects risk perception among its practitioners and produce fears, dissatisfactions and thus may stop the use of DevOps. Consequently, this reinforces the relevance of our question whether the practice of agile to DevOps generates job dissatisfaction or satisfaction. According to Herzberg (1964), a job can generate both satisfaction and dissatisfaction at the same time but not due to the same factors. Achievement, intrinsic interest in the work, responsibility, and advancement would contribute a lot to job satisfaction while company policy and administrative practices, supervision, interpersonal relationships, working conditions, and salary would participate very little to job satisfaction, but contribute to dissatisfaction (Herzberg 1964). Therefore, although generally agile adoption is a source of job satisfaction, it could as well lead to job dissatisfaction. We also assume the same reasoning for DevOps as generator of satisfaction and dissatisfaction. Solinski and Petersen (2016) investigated agile practice usage and its benefits and limitations and they found that agile can generate dissatisfaction within agile teams. Frequent deadlines and highly ritualized and sequenced agile methods increase stress levels of employees (Petersen and Wohlin 2009). In addition, communication issues may produce dissatisfaction. Agile could actually and paradoxically be fatal for communication with some agile teams potentially isolated from their environment (Karlström and Runeson 2005) or with difficulties to communicate externally for agile teams (Pikkarainen et al. 2008). Melnik and Maurer (2006) identified some reasons for dissatisfaction among employees working in agile, i.e. incompetent management, unclear company’s objectives. Thus, we consider that DevOps, as an extent of agile, can be a source of job satisfaction as well as job dissatisfaction.

### ***Job Satisfaction Through IT Job Design***

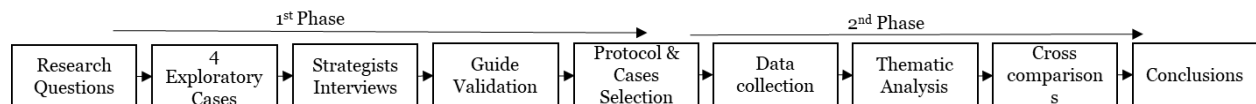
We retain for our study Locke (1976, p.1300) definition of job satisfaction defined as “a positive or pleasurable reaction resulting from the appraisal of one's job, job achievement, or job experiences”. In order to assess job satisfaction, we selected job characteristic theory (Hackman and Oldham 1980). This theory is part of job design approach and propose to design 5 core job characteristics (CJC) in order to generate individual satisfaction, global satisfaction and hence organization’s performance. In their

framework, Hackman and Oldham (1980) explain these five CJC: 1-skill variety, defined as the extent to which the job requires the job-holder to use different skills and abilities and knowledge; 2-task identity, characterized as the extent to which a job enables the job-holder to achieve a whole identifiable outcome; 3-task significance, described as the extent to which a job influence and impact organization and people's lives; 4-autonomy, determined as the extent to which a job provides the job-holders a feeling of freedom, giving them the possibility to schedule their activities and the way to do it; 5-feedback, qualified as the extent to which job-holders receive clear information about their performance. These five CJC generate three critical psychological states in job-holders: 1-meaningfulness, the feeling to contribute, to add value, to be important; 2-responsibility, the degree to which the job-holder feels accountable for the job done, outcomes; 3-knowledge and understanding of the results by each job-holder to perform and improve their job. Then, these three psychological states enable job-holder high intrinsic work motivation, high growth job satisfaction, high quality of work, effectiveness and performance, while reducing absenteeism and turnover. To implement work redesign and improve job characteristics, Hackman and Oldham (1980, p.135) propose five principles: Combining tasks; Forming natural work units; Establishing client relationships; Vertically loading the job; Opening feedback channels. These principles seem to be tightly related to DevOps pillars and more specifically to Culture and Sharing. Consequently, considering these principles, we may assume that DevOps has a theoretical and/or practical impact on job characteristics. Therefore, DevOps could be an enabler of work redesign, hence of job satisfaction. IS literature used widely job design framework applied to IT teams. In particular, many authors noted that significant changes on job environments could influence of job characteristics and their impact on job satisfaction, hence on results, performance (Keil et al. 2000; McKnight et al. 2009; Thatcher et al. 2002; Tripp et al. 2016). Tripp et al. (2016) found, through a quantitative study of 252 agile and non-agile team members (mostly project managers and developers) and using the job characteristics model, that the use of agile practices positively impacted job satisfaction within team members. Consequently, and regarding our previous literature review, following agile logic, we suggest that DevOps philosophy and practices noticeably impact job characteristics and thus job satisfaction. However, as for any management method, pushing the envelope too far may also result in risks and dissatisfaction. Consequently, we analyze the perception of how DevOps may lead to further improvements in job satisfaction or dissatisfaction by taking into account these other factors in an in-depth qualitative study.

## Methodology

### Design overview

We conducted this research in 2017 at Group X, a very large European firm in the services industry. IT Staff represent 15000 people. They used agile methods since more than 15 years, mostly Scrum and XP, and DevOps for 8 years. Being one of the early adopters of the DevOps approach, the company tested in two ways the adoption and implementation of DevOps for a few years on pilot projects. First, the firm went through a gradual transition from a software development context working in agile to a DevOps context; Secondly, by starting a project directly following the DevOps approach. Then, in January 2015, the firm launched a strategic plan to broadcast DevOps at the group level. Thus, this DevOps firm's experience provided a revelatory case study context (Yin 2013) in which job satisfaction factors and risk perceptions were particularly likely to appear when adopting DevOps. To answer to our research questions, we followed an exploratory inductive embedded case study research methodology integrating interviews, observations and documentation (Eisenhardt 1989; Yin 2013). We conducted the research into two phases (cf. Figure 1).



**Figure 1. Case Study Research Design**

In Phase 1, we followed five steps to select job roles to be interviewed in a sample of projects: 1- Scientific literature review to get familiar with knowledge about DevOps and related perceptions and formulate research questions; 2- Full immersion within four DevOps teams for 15 days to observe and interview them to compare elements identified from the literature and those present in practice, 3- Interviews with

12 strategists from Group X to incorporate their vision and to validate sample. We selected five job profiles: two people from the management, e.g. Product Owner (PO) and Department Manager (DM); two people in operations, e.g. Production Engineer (PE) who perform tests as Ops (developers also perform tests in a DevOps mode) and Release Manager (RM) and one person in design (Architect, AR). 4- To ensure validity of the interview guide, the latter was submitted to strategists of Group X and to several researchers. Then, the guide was pretested with five selected profiles and was adjusted, i.e. one question has been simplified and two new questions were added. 5- Three criteria were retained to select the sample: 5.1- the size of the project measured in terms of number of staff assigned to the project (Rolland et al. 2016) 5.2- the degree and nature of outsourcing on the project, fixed-price contracts could be problematic when working in agile mode. 5.3- the level of maturity in terms of the agile to DevOps transition as described below. All theoretically possible configurations gave us 12 different types of projects to identify with 5 profiles per project. This research covers 11 projects with a complete analysis done.

In Phase 2, we interviewed these 5 job roles among 11 project teams, which involved 54 in-depth face to face individual interviews in the company. We performed a thematic analysis and theorized to deliver conclusions.

**Data collection**

We collected data over a period of eight months, from January to August 2017. Data, documents, observations and interviews performed in the first phase considerably helped us in interpretation of the 54 interviews in the second phase (Table 1). This use of multiple sources of evidence corroborated findings and strengthened validity and credibility (Yin 2013).

Phase 1 Sampling and Questionnaire Building - Collected data	12 open in-depth interviews (12 strategists, e.g. HRD, CIO, Head of Marketing, etc.) recorded and lasted from 45 to 120 minutes
	22 semi-structured interviews with Biz, Dev and Ops profiles, e.g. Product Owners, Technical and Functional Architects, Department Managers (Dev or Ops), UX Designers, Technical Leaders, Developers, Scrum Masters, Testers, Production Engineers, Release Managers, etc. Recorded and lasted from 30 to 140 minutes.
	4 focus group interviews (partially recorded and lasted from 40 to 120 minutes)
	1 DevOps day (full DevOps team present): 380 minutes recorded
	Documents collected from the firm (Business repository, company agreements, etc.)
	Field notes, photos, maps
	5 interviews testing the questionnaire (recorded, lasted from 60 to 90 minutes)
Phase 2 Data Collection	54 recorded interviews with DM, PO, AR, PE and RM. Each interview was recorded, transcribed and lasted 90 minutes on average

**Table 1. Data Collection per phase**

Phase 2 interviews were conducted by three interviewers. For a better reliability, codification was triangulated at least by two researchers and codes were consistent and similar. Finally, to strengthen reliability, we followed our case study protocol and we developed on a platform a case study database (Yin 2013).

**Data Analysis**

Our thematic analysis of the discourses followed the six-phase framework provided by Braun and Clarke (2006) that offers an accessible approach for analyzing qualitative data. Using this clear guide as a method, we identified important themes. As Braun and Clarke (2006) explain, a theme is a pattern that captures something significant or interesting about the data and/or the research question. Themes emerged that represent factors of satisfaction, dissatisfaction, perceived risks and difficulties At the fifth step (defining and naming themes) of the six-phase Braun and Clarke framework (2006), we considered the themes themselves, and each theme in relation to the others. To help understanding how factors interact and relate to each other’s, especially between the sources of dissatisfaction and perceived risks and difficulties, we wanted to identify the causal links that individuals had drawn between these themes.

For the present research, we focus on the criteria of agile maturity and consider agile (noted AG) versus DevOps (noted DO). AG maturity level in the transition to DevOps is reached when teams utilize agile

methods or develop an hybrid one (Fitzgerald et al. 2006). Agile teams expedite frequent releases because development becomes more iterative. However, DevOps is not achieved because silos between Dev and Ops still exist and limit sharing. Common culture, automation of releases or common useful metrics and full speed are not reached. DO maturity level refers to the use of Continuous Integration. When practicing continuous integration, the Ops function is aligned with the Dev function. Both perform various tests (unit and non-regression tests) (Ståhl and Bosch 2014) which are synchronized with code development. When moving to Continuous Delivery / Deployment, integration tests, end-to-end tests, performance tests, user acceptance tests are performed by Ops and co-designed with the Dev function (Chen 2017).

## Findings

Our qualitative method allowed us to identify factors common and specific across projects in agile mode and in DevOps mode (Table 2). Due to length limitations and given the lack of elements on DevOps in IS literature, for the common factors to DevOps and agile, we use illustrations coming from DevOps teams in a first sub-section, before turning to those that are specific to agile mode in the second sub-section.

### **Satisfaction, Dissatisfaction, Risks, Difficulties for DevOps Teams**

#### **Satisfaction**

Our thematic analysis of individuals discourses in teams revealed ten job satisfaction factors for most of the five studied profiles. We present them and associate a verbatim illustrating each factor. **1- Task variety.** DevOps methodology has satisfied several interviewed individuals, who appreciate testing varied teamwork methods, several tools and technologies, or benefiting from a daily learning mode in their profession. This attraction for the tasks variety is felt by the different studied professions. A release manager (RM-DO): « *You see the business needs, you see the advances and all the techniques that are implemented ... We still touch a lot of areas, so it's an interesting job for that.* » **2- Freedom of action and autonomy to experiment.** Individuals appreciated the autonomy and the freedom of action related to the DevOps methodology. PO's in particular are sensitive to the freedom of project planning, or to the authorization to experiment with the project team. This experiment is allowed since the correction of an error can be done quickly or another experiment can be conducted. A development manager (DM-DO): « *There is an experimental side, because we test things, it works, it doesn't work, we like it, we don't like it; Agility allows you to have autonomy, because you can quickly prove that you are in the right direction.* » **3- The role to play in a collective.** New PO and RM roles have emerged with the DevOps approach, which brings them satisfaction. The other interviewed key roles also appreciate the place the DevOps approach places on their role within the project team, giving them the feeling of having contributed their expertise to the project. A PO-DO: « *Being an orchestrator and taking a team to a project, to a shared goal, to make sense.* » **4- A recognized role.** Some appreciated the recognition of their role in the eyes of other team members; this is particularly the case for AR and production engineers (PE). A PE-DO: « *We are more recognized than before and now there are more exchanges, there are more relationships; we went from being a performer to a partner position.* » **5- Participation in a successful project.** The contribution to a successful project brings satisfaction to several individuals, who take pride in it. A RM-DO: « *What is satisfying is taking a project into production and once in production, there are no incidents. That's it, we think we worked rather well.* » **6- Feedback from the team and the client related to the work done.** Some appreciate the recognition of users, or even of customers, giving appreciable feedback on the quality of work accomplished. A PO-DO: « *I'm satisfied when users tell me that they are happy or when I go to the store and that people say: ah it's fine to have this app!!* » **7- The richness of the collaboration between roles.** All interviewees highlighted enhanced collaboration within the team as the main factor of satisfaction. The effectiveness of this collaboration is evoked, since everyone is free to express himself and problems are debated together to find solutions. The information is shared within team members, which allows to understand the constraints of different members. An architect (AR-AG): « *All trades are respectable for what they are and there is really a tangible proof. We know what everyone is doing, it makes it easier to pass information and above all it avoids the problems.* » **8- The coverage of a market need and interest in projects.** The noted satisfaction also concerns the variety of projects or the project themes related to the strategic orientations of the company, or the response to a customer need. A RM-DO: « *I like to know that in the end, the customer will have a new service and that it will meet one of his needs.* » **9- The**



**collective ability to collaborate and group dynamics.** Through the DevOps approach, the teams seem unified and the terms « tightly knit », « good », « competent », « friendly » or « good energy » are evidence of this. This shows a certain collective capacity to collaborate, which gives meaning to the action and brings satisfaction to the team members. A product owner (PO-DO): « *We have a certain agility to solve unforeseen problems that happen every time. So, in general, we manage to respond to what we are asked even if it happens quite late or if it was not planned.* » **10- The collaborative process, allowing swift continuous improvement.** PO's, managers and RM emphasize the responsiveness of their team, which resolves problems more quickly, as well as upstream exchanges that can be quickly alerted to a risk and change the approach in the next sprint. A RM-DO: « *We can be alerted earlier to raise the alert on the developers' side, so they think of some corrections. There is still time to postpone an edit in the next sprint* »

## Dissatisfaction

We now present the results identifying six dissatisfaction factors from individuals within teams. **1- Administrative process red tapes.** Many individuals have criticized the administrative process, as well as the many tools to use for reporting. Some believe that the structure is not optimal for working with agile or DevOps practices, or that this complexity leads to inertia for the organization. The set of processes to respect and the tools to use slow down the pace of project management and do not allow to fully control the schedule, according to a manager. These daily difficulties during DevOps project management are dissatisfaction sources. A PE-DO: « *We have an accumulation of work tools, of processes, of interfaces to use. As soon as we do an action, we have to update data on a tool, so there are a lot of cross-functional operations... which makes it rather complex* » **2- Reduced time and lack of depth of work.** Work overload and lack of time are factors of dissatisfaction and imply a feeling of lack of deepening of the works. Lack of time is criticized, preventing sufficient study of the application, carrying out all the tests or treating the tasks exhaustively. A production manager (PM-DO): « *Deployments too fast: it's sometimes frustrating to have delivery dates imposed on us without having properly or correctly done our jobs.* » **3- A geographical distance hindering collaboration.** The geographic distance between team members is also criticized, as budgets do not allow the physical meetings of all members for recurring meetings. An AR-DO: « *Typically the lack of colocation of teams, but we can't always answer this because of lack of budget with fairly distributed teams.* » **4- Roles exposed to criticism.** RM and PO's express some dissatisfaction with the roles they play. They sometimes feel judged by others in their innovative project management approach, feeling that they have to justify their choices. A PO-DO: « *Sometimes I feel that I have to justify our choices, and finally the methodology, the impression that we are judged in fact.* » **5- Operator overload (project and incident management).** Some production people also complain about the overload of work in their department, since they are responsible for DevOps projects and incident management to be conducted in parallel. A PE-DO: « *We have applications on which we are still operating level 1. We do a lot of things at once, there are tasks on which we would like to focus ... sometimes we are a little disturbed by the incident.* » **6- Lack of feedback from the hierarchy.** DevOps teams are composed of members of various trade and has an own operating mode that is independent of the hierarchical lines and traditional craft. Individuals regretted not having enough feedback from their supervisors and regretted this lack of attention. A DM-DO: « *My hierarchy is not very present, so I don't have too much feedback from my boss.* »

## Risks and Perceived Difficulties

Our thematic analysis highlighted eleven factors that we present below. **1- Resistance to change and a fear for the future of the operators.** First, among the barriers to DevOps membership, resistance to agile transition is especially high for employees with an extensive experience in plan-driven methods. DevOps team members insist on the necessary personal desire to work closely in collaboration to get involved in the process. There is also a concern about roles and the future of production engineers, due to automation by the developers and brought about by these new methods. A PE-DO: « *The main risk for me is that production is exclude from the process, so we do DevOps without operations.* » **2- Increased work cadences, resulting in work overload and tasks prioritization:** the increased work rates and the related workload entail human and quality risks. The tasks prioritization is mentioned, with the risk of neglecting some, such as ensuring security fundamentals (emphasized by the AR) or conducting a systematic qualification. Moreover, lack of time to step back or train may result in exhaustion. A PO-DO:

« People are still under pressure all the time, I think this is one of the big risks besides agility, it is a risk of team members exhaustion, because we go from sprint to sprint constantly, with always extremely high pressure, without enough time to have a break. » **3- Rushed project launches and post hoc corrections.** Many interviewees mentioned the risk of embarking on a rush in the process, requiring some elements to be corrected ex post. A RM-DO: « The sprint schedule should not be an excuse for pushing through the preliminary analysis. Sometimes I think it's useful to have one that is really clean to avoid spending time while doing, redoing, undoing, redoing... » **4- A loss of global vision of the project.** Another risk factor concerns the risk of loss of overall vision of the project, that is divided into sprints. Individuals consider that it is essential to ensure its overall consistency as well as its compatibility with other projects. A PO-DO: « By dint of divide and formalize everything, we are concentrated on the current sprint. Or even we follow the previous and the next, but don't we lose a little of the whole vision? » **5- Unsuitability of project for DevOps practices.** Several individuals also evoke a risk of DevOps fashion effect: some projects are managed in this way whereas a classic cycle would be just as relevant. This is particularly the case of too complex projects, which would be difficult to break into sprints. A manager-DO: "I think we need to evaluate the need to adopt the approach. The risk is to launch a fashion effect: to make DevOps to make DevOps. " **6- Risks hidden by automation.** Individuals evoke risks that may remain hidden by automation and make more complex the application maintenance. A PE-DO: « With automation, we have less and less knowledge about the app. Today with automation, I don't know what has changed and I have no vision of what has been done. And besides, the applications with [functional] problems are often the applications that seem to work very well. » **7- Lack of coaching.** Some actors highlight the need for coaching in the DevOps approach for team members. A PE-DO: "On this project, we are followed by a DevOps Coach who helps us to make meetings, to ask questions, to put a plan, to see the different steps, how we could evolve ... Without the coach I think the transition is going to be a bit difficult because it's not yet natural. « **8- Planning constraints in case of new project.** Risks can be related to planning management, because it is difficult for individuals to simultaneously manage several DevOps projects due to the important involvement required. Since individuals are strongly mobilized by this method and the implementation dates, it can be difficult to engage the team on a new project, which also includes firm dates and a customer's expectation. These constraints are difficult to reconcile at the operational level, particularly by PE's, and require knowing how to manage priorities at the level of the PO. A PO-DO: « Are there any things planned for next month or the next quarter that we have to stop to look into this new opportunity? So, change management is still a big problem. » **9- Administrative process red tapes.** Some of the interviewed individuals have difficulties with administrative processes red tapes. Indeed, several interviewees insisted that DevOps project management is time-consuming and costly in human resources mobilizing. The process is timed by numerous meetings and obligatory passages in committee before going into production, which finally and paradoxically induce a certain slowness in the process of project management. These difficulties are found in the sources of dissatisfaction already noted in the speeches. A manager-DO: « We have processes that prevent us ... from being agile precisely, from being able to react quickly, efficiently. For example, today we have a committee that is held on Monday, which must allow us to put into production. So, we have to register, I believe before Thursday, so ... if we register on Friday, we have ten days of time to put in production. »

### **Specific Dissatisfaction and Risks for Agile Teams**

Moreover, our thematic analysis of individuals discourses in Agile teams revealed three specific elements to this project management type, which did not appear in the discourses analysis of DevOps teams. A dissatisfaction factor concerns an unfulfilled expectation of even closer collaboration: some Ops reckon that they can share more problems they encounter thanks to this Agile open discussion space, but at last expect the Dev to stay really tuned to them. A RM-AG: "What may not satisfy me is sometimes the relationships with the Dev: they do not listen to us, because the voice of the run is not important, whereas it should be!" We also revealed two specific Agile risks. First, a need for collective acceptance of practices in the team: some people evoke resistance to change from staff used to IS plan-driven projects management practices. This is particularly true since the company's workforce is aging. A manager-AG: "It is difficult for somebody who is managing plan-driven projects for thirty years; one has to change everything and to be more autonomous... It's a big change." Second, simultaneous agile or plan-driven projects management practices: calendars are different, as are response times, tools and processes, and that involves for some Ops time and organization constraints. An AR-AG: "I also have a non-Agile project

and therefore it is very complicated. It's as if you had to be able to speak four or five languages at the same time, to have rhythms of life that were different depending on the people you are talking to.”

**DevOps Brings More Satisfaction than the Agile, but Remains Riskier**

Common factors	Specific factors to one mode of collaboration
Satisfaction	
Task variety Freedom of action and autonomy The role to play A recognized role Participation in a successful project Feedback from the team and the client related to the work done Richness of the collaboration between roles Coverage of a market need and interest in projects	DevOps practices: group dynamics and the collective ability to collaborate DevOps practices the collaborative process, allowing work improvement (service)
Dissatisfaction	
Administrative process red tapes Reduced time and lack of depth of work Geographical distance hindering collaboration	DevOps practices: roles exposed to criticism DevOps practices: operator overload (project and incident management) DevOps practices: lack of feedback from the hierarchy Agile practices: an unfulfilled expectation of even closer collaboration
Risks and perceived difficulties	
Acceptation obstacles and a fear for the future of the operators Increased work cadences, resulting in work overload and tasks prioritization Rushed project launches and post hoc corrections A loss of global vision of the project related to its split in multiple sprints Unsuitable project for agile or DevOps practices	DevOps practices: risks hidden by automation DevOps practices: lack of coaching DevOps practices: planning constraints in case of new project DevOps practices: administrative process red tapes Agile practices: a need for collective acceptance of practices in the team Agile practices: Simultaneous agile or plan-driven projects management practices

**Table 2. Agile and DevOps comparison**

**Satisfaction**

By observing the satisfaction factors from our thematic analysis, we find that the factors are overwhelmingly common to both modes of collaboration (8 of the 10 factors are common). Indeed, we find the same satisfaction factors related to the richness of the collaboration in the project team (introduced and innovative with agile practices, or deepened with DevOps practices), new and recognized roles (agile practices) or enriched roles (DevOps practices), the attraction for the new methodology, involving autonomy and freedom of action (as a "start-up spirit" according to a manager), the participation in a successful project, or feedback from team members or clients on the work done. However, the analysis of projects in DevOps mode shows a certain progression in the individual satisfaction, on the one hand because of more numerous sources of satisfaction and on the other hand, because the implication in the approach appears more profound. We discover the attraction for daily learning and for swift continuous improvement, thanks to the collaborative process. Individuals can not only invest to improve the team practices and "think of new organizations" (a RM), but also to bring out new solutions that satisfy the various team members. While in the agile method, there is an attraction for collaboration and exchange in the project team, the DevOps projects show a sense of existence of a collective capacity to collaborate within the team. PO's, managers and RM emphasize the responsiveness of their team, which resolves problems more quickly, as well as upstream exchanges that can be quickly alerted to a risk and change the approach in the next sprint. Through the DevOps approach, teams seem to be unified and the terms "welded", "good", "competent", "friendly", or "good energy" are evidence of

this. Not only are some PO's, PM's or managers satisfied with such a team relationship, but they feel that the team is also satisfied and motivated to continue this work together. This shows a certain collective capacity to collaborate, which makes sense of the action and leads to satisfaction for the team members.

### **Dissatisfaction and risks**

By contrast, the second and third part of the table presents a different configuration of the factors: some dissatisfaction factors and perceived risks or difficulties appear to be common to both modes of collaboration, but more DevOps specific factors appear in the right column. Indeed, in agile mode, there is only one specific dissatisfaction factor and two specific risks or perceived difficulties, while for projects in DevOps mode, we note three specific dissatisfaction factors and four specific risks or perceived difficulties. In agile mode, these specific dissatisfactions, risks and difficulties mainly concern a lack of effective collaboration within team members (an unfulfilled expectation of even closer collaboration, and a need for collective acceptance of practices in the team) that generates frustration while the move to the agile should allow them to become more involved in project management decisions and to further exchange ideas on the encountered difficulties. The second difficulty emerges from individuals with plan-driven and agile projects in the same project portfolio, which requires them to use different ways of working, collaboration modes and tools at the same time (simultaneous agile or plan driven projects management practices). On the contrary, in DevOps management projects, several specific dissatisfaction factors and risks have been identified. These different elements can be grouped into four sub-themes. 1- *More exposed roles*: Indeed, the DevOps roles are more exposed in the project team and can be exposed to criticism to justify their experimentation (roles exposed to criticism). This factor of dissatisfaction hinders the overall satisfaction felt by a majority of team members about the role to play and its recognition. 2- *Lack of organizational support*: Since the DevOps project team is multi-line of services, the superiors have no control over the team and may seem overwhelmed by this cross-functional mode of operation (which generates a lack of hierarchical feedback). In addition, the approach is innovative in the way the team operates and a coach appears essential to help members to self-organize (lack of coaching). 3- *A complex time management*: The DevOps approach requires a very high level of availability and involvement on a daily basis, which makes it difficult for the operators to manage projects and incidents concomitantly (operator overload). The schedules are also tight and planning a new project becomes very difficult (planning constraints in case of new project). Finally, with this difficult time management and while the approach advocates autonomy and freedom of action, the respect of administrative process steps present a real daily constraint that must be alleviated (administrative process red tapes). 4- *Automation risks*: In terms of risks, it seems that automation can hide some functional defects in the developed application, which PE's will have difficulty maintaining later (risks hidden by automation). In summary, our findings suggest that the DevOps approach brings greater sources of satisfaction to team members, but also represents factors of dissatisfaction and perceived difficulties and potentially hidden risks. While potentially more satisfactory DevOps is also riskier if investments in time, hierarchical resources and automation are insufficient.

## **Discussion**

We deepen our results and discuss them in two points. We first analyze the correspondence of satisfaction factors emerging from the discourses with the five CJC of Job Design by Hackman and Oldham (1980). Then, we study the links between the factors that the individuals interviewed have woven themselves to understand the relationships between these categories.

### ***DevOps Satisfaction Factors and Core Job Characteristics***

Based on the satisfaction factors highlighted in our discourses thematic analysis, we found ten satisfaction factors that were presented previously. Depending on their significance, we find a correspondence with the five principles of Job Design highlighted by Hackman and Oldham (1980) reviewed in literature. We present this comparison in the table below (Table 3). This reconciliation has already been done by some authors for agile projects (Tripp et al. 2016), but not for DevOps projects. As an extension of these studies, the correspondence observed here reflects the fact that the deployment of the DevOps approach brings satisfaction for the concerned job-holders. While the Hackman and Oldham (1980) CJC's framework helps to see convincingly why DevOps teams can experience high satisfaction levels, some aspects of the DevOps phenomena can be further explained by other theorization of work processes at team level. For

instance, autonomy experienced by individuals in DevOps and agile teams is related to task variety but also to sequential variety (Pentland 2003). Such variety appears to provide an index of tacit, procedural knowledge, while task variety may relate more closely to explicit, declarative knowledge. With a wider scope of activity, it can be argued that sequential variety of DevOps team is also larger than that of agile teams. In addition, sequential variety can contribute to both richness of collaboration and to finding better solutions which participates to the construction of a collective and to grow dynamics, all elements that are also related to skill variety.

CJC	Definition (Hackman & Oldham 1980)	Satisfaction factors
Skill variety	<i>“the degree to which a job requires a variety of different activities in carrying out the work, involving the use of a number of different skills and talents of the person”</i> (p. 78)	The task variety The richness of the collaboration between roles The construction of a collective and group dynamics The collaborative process allowing swift continuous improvement
Task identity	<i>“the degree to which a job requires completion of a “whole” and identifiable piece of work, that is, doing a job from beginning to end with a visible outcome”</i> (p. 78)	The role to play in a collective Participation in a successful project
Task significance	<i>“the degree to which the job has a substantial impact on the lives of other people, whether those people are in the immediate organization or in the world at large”</i> (p. 79)	A recognized role Coverage of a market need and interest in projects
Autonomy	<i>“the degree to which the job provides substantial freedom, independence, and discretion to the individual in scheduling for the work and in determining the procedures to be used in carrying it out”</i> (p. 79)	Freedom of action and autonomy to experiment
Feedback	<i>“the degree to which carrying out the work activities required by the job provides the individual with direct and clear information about the effectiveness of his or her performance”</i> (p. 80)	Feedback from the team and the client related to the work done

**Table 3. DevOps Satisfaction Factors and Core Job Characteristics**

### ***Relations Between Satisfaction Factors, Dissatisfaction Factors and Perceived Risks***

Certain perceived risks and difficulties may lead to dissatisfaction with the individuals interviewed, while others do not seem to have any influence on the dissatisfaction expressed. Finally, some sources of dissatisfaction may decrease the satisfaction felt on the same theme, while some suggestions for improving working conditions may increase satisfaction on certain points.

### **Risks and Perceived Difficulties Constituting a Dissatisfaction Source**

*Agile and DevOps practices: rituals and tools to be simplified.* The agile or DevOps practices implementation requires to follow defined steps, so to respect some process. Yet, globally, individuals in all trades have expressed dissatisfaction on this point. Their concerns bear on the administrative red tapes, imposing recurring reports, managed on different tools, and presentation in successive committees. The time spent on these compulsory steps is experienced as a source of unduly constraints, especially since they seem paradoxically to slow the DevOps project management progress. The tools are also adaptable, because sometimes not very ergonomic, and limit in number, because there are multiple, and practices are not unified in the teams. All roles also advocated simplifying these processes to make them more agile. Thus, the very existence of an agile process to respect is not criticized because it borders the

individual actions. However, its complexity is called into question because it appears time-consuming, slows down the project progress and is a source of dissatisfaction. A simplification is unanimously demanded because it would bring more freedom of action (autonomy CJC).

*To Optimize time spent on project and to reinforce operators' skills.* The division of a project into parts, managed in sprints, allowing to produce each element more quickly and to deliver the solution faster to the customer, leads to satisfaction for individuals when the solution is delivered without major difficulty or quality problem. The cycles speed also creates drawbacks, potentially affecting the work quality or the solution put into production. Indeed, some steps such as test phases can be lightened to meet deadlines, some tasks may be managed in priority over others and undetected risks may remain, due to automation. These risks of lack of quality create dissatisfaction and additional difficulties for PE's who are responsible for solving the latest problems. Moreover, high cadences for the individuals can cause tension between members, stress and difficulties to organize everything, that represents dissatisfaction factors (Shropshire et al. 2017). These rates may carry risks and difficulties because these sustained rhythms are difficult to hold and the risk of exhaustion may appear. According to the interviewees, it seems important to provide rest periods between each deadline. Free time can also help to take a step back from the project as a whole, or to monitor training and increase skills. We find here one of the obstacles of the agile transition, identified by Ghobadi and Mathiassen (2016), on a tight sprint schedule with little time for interaction. To solve these problems, some interviewees propose a new structure for operating team, with DevOps specialists. This could allow the operators to assert themselves in front of developers and to affirm the collaboration between the two business lines. In summary, the agile cycles speed creates difficulties and risks to the service offered quality, which tarnishes the satisfaction felt by most roles on the work done. Suggestions for the reorganization of the operating teams are proposed, relying in particular on a strengthening of the cross-functional collaboration and a rise in skills (skill variety and task identity CJs). Moreover, it seems essential to try to free up time in the projects management, not to put in suffering the individuals but to allow them to realize themselves and build skills. We find here the need to cover the factors of hygiene, maintenance or atmosphere, necessary for the good mental balance of the individual (Herzberg 1964), which, according to this theory, if they are not covered, will lead neither to satisfaction nor motivation but to dissatisfaction.

*Support the motivation and cohesion of the project team.* The move to agile methods and DevOps is a real organizational change, placing human resources at the heart of the concerns. Thus, the company must support and respond to the need for acculturation, training and motivation of individuals in the process. According to our results, conviviality and group dynamics are factors of satisfaction, their motivation could be reinforced if hierarchy was able to keep up with the pace of change and gave appropriate feedback (feedback CJC). Change can also be more difficult to implement in geographically dispersed teams. The distance does not facilitate team building and this dispersion appears as unfavorable (although not an impossible barrier to overcome) to agile and even more to DevOps practices. The different business lines individuals feel either dissatisfaction or difficulties for an effective agile approach in their project team. An advanced solution is thus to bring together the members of the project teams geographically and to create conviviality moments outside the office to bring cohesion to the group (Deshpande et al. 2016). Change management involves human management, an acculturation that requires time and organizational efforts to bring team members together to reinforce group dynamics and the collective capacity to collaborate (skill-variety CJC).

### **Decrease Certain Factors of Dissatisfaction Can Increase Satisfaction**

*Strengthening the positioning and support of central roles.* Agile and DevOps practices allow to coordinate cross-functional collaboration. The PO and RM central roles respond to this need. The roles allocated satisfy the individuals on the aspects of communication and information sharing with the teams, but they are also some dissatisfaction factors, because they sometimes feel judged on their choices in particular. Their perceptions of the role difficulties come together, because of their cross-functional and non-hierarchical positioning. Their position imposes a "dual hierarchical structure" on the job and project team members, leading to uncertainty and difficulty in positioning these roles. Team members playing these roles want them to be more framed and explicit, that their position be upgraded with definition of roles and responsibilities scope. They also expect exchanges of best practice between project teams, even the appointment of an agile coach for each of the two roles. The accompaniment of these two central roles

is therefore desired to support the smooth operation of the project team and cross-functional collaboration (task significance CJC).

### **Strengthening Some Satisfaction Factors Can Enhance Satisfaction**

*Strengthening collaboration between team members.* This point concerns one of the central principles of agile approaches, the collaboration development between team members. Everyone can benefit from this wealth of openness to others which allows to better understand the upstream and downstream constraints of the collaboration chain and the broader context of the project. We thus join the second CJC of varied and coherent set of tasks, proposed by Hackman and Oldham (1980). Thus, improvements are suggested to encourage and strengthen this collaboration. POs and managers campaign for a unified team managed by a single supervisor, for which annual and collective goals will be set. This recommendation is consistent with the idea of reinforcing the feedback CJC of Hackman and Oldham (1980), requiring areas for results monitoring and quality control. These collective objectives would make the entire project team more accountable to the quality of the overall process and product service. In addition, when the team is formed, some would like to keep members for future projects as work habits are taken, which makes it more efficient (according to DevOps RM). Finally, exchanges of good practices could be organized between teams, to contribute to the continuous improvement of team practices. Thus, the principle of collaboration between professions is presented as a major source of satisfaction for individuals (Hackman and Oldham, 1980). It deserves to be consolidated between Dev and Ops members, even extended beyond the scope of the project team.

## **Conclusion**

Knowledge is scarce regarding risks and difficulties faced by organizations face when moving to DevOps. In such transition, this knowledge is important both to facilitate the IS function transformation and to create value.

Consequently, this paper sought to investigate two questions. Firstly, we investigated which difficulties and risks were perceived by employees using DevOps and job (dis)satisfaction among employees using DevOps. We found that DevOps brings more factors of satisfaction than dissatisfaction, although it comes with several risks and challenges. Our findings regarding the link between job design and satisfaction are consistent with Hackman and Oldham's CJC for DevOps team and tend to justify the move to DevOps.

Secondly, our paper investigated if there were any differences of perception about risks, difficulties as well as job satisfaction between Agile and DevOps approaches. We showed that DevOps can increase job satisfaction in comparison to agile since agile teams expect closer collaboration with operations and since DevOps members display additional factors of satisfaction related to group dynamics and collaborative process. This result is coherent with Wiedemann (2017) who highlighted the importance of these new collaborations when adopting DevOps. However, DevOps also leads to more diverse types of risks in comparison to agile. We explain this result firstly because DevOps enables practitioners to open widely their eyes and hence to detect risks beyond silos; secondly because of the working conditions specific to the analyzed firm and maybe not present in another firm. Increased risks perception could generate more fears, more resistance to change and inertia (Besson and Rowe 2012). We showed that DevOps fosters job satisfaction and these findings differ clearly from those of Shropshire and Sweeney (2017) research where they identified that job satisfaction was declining following DevOps implementation. In addition, we highlighted that DevOps brings more satisfaction than the Agile but remains riskier which could appear ambivalent.

One limitation of our study is related to the choice of our methodology: an inductive exploratory case-study based on a rich sample of projects within a single large company. However, future research could investigate these issues in a more quantitative approach or across a wider sample of organizations for a statistical generalization and take into account change management strategies highlighted by Chen (2017). Another limit is related to the structure of the organization we analyzed with two main distinctive hierarchical lines, i.e. Dev and Ops, and with teams working remotely. Such conditions were not favorable to DevOps. Comparisons we made with other firms showed that when there is a unified hierarchical line with a common dashboard, feedback from the hierarchy was much more important. Under such conditions, feedback characteristic (Hackman and Oldham 1980) would be higher and job satisfaction like

performance would probably increase. As a result, we can interpret the studied case as a conservative case. A third limit is the use of participant observation during phase 1 due to the presence of the scientific team. Finally, positive outcomes such as job satisfaction may be related to other factors such as work conditions (Herzberg 1964) and future research is needed to confirm our initial findings.

Our study contributes to the understanding of DevOps through a deep insight within DevOps teams. We need more studies to understand precisely mechanisms and effects of DevOps. Skill variety CJC should be investigated more deeply. Hackman and Oldham (1980) explained the necessity to experience, to explore and manipulate the environment in order to gain sense of efficacy using skills. Those are mechanisms of learning (Zollo and Winter 2002) and DevOps seems to amplify these learning mechanisms through continuous learning and continuous experimentation as recommended by Kim et al. (2016). This amplified learning through DevOps could be a vector of organizational innovation (Cockburn and Highsmith 2001; Highsmith 2009) and could be as well a source of dynamic capabilities.

## References

- Acuña, S. T., Gómez, M., and Juristo, N. 2009. "How Do Personality, Team Processes and Task Characteristics Relate to Job Satisfaction and Software Quality?," *Information and Software Technology* (51:3), pp. 627–639.
- Albadarneh, A., Albadarneh, I., and Qusef, A. 2015. "Risk Management in Agile Software Development: A Comparative Study," in *Applied Electrical Engineering and Computing Technologies (AEECT), 2015 IEEE Jordan Conference On, IEEE*, pp. 1–6.
- Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern Jon, Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. 2001. *Manifesto for Agile Software Development*.
- Begel, A., and Nagappan, N. 2007. "Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study," in *1st International Symposium on Empirical Software Engineering and Measurement (ESEM 2007), Madrid, Spain: IEEE*, pp. 255–264.
- Besson, P., and Rowe, F. 2012. "Strategizing Information Systems-Enabled Organizational Transformation: A Transdisciplinary Review and New Directions," *The Journal of Strategic Information Systems* (21:2), pp. 103–124.
- Boehm, B., and Turner, R. 2005. "Management Challenges to Implementing Agile Processes in Traditional Development Organizations," *IEEE Software* (22:5), pp. 30–39.
- Braun, V., and Clarke, V. 2006. "Using Thematic Analysis in Psychology," *Qualitative Research in Psychology* (3:2), pp. 77–101.
- Chen, L. 2017. "Continuous Delivery: Overcoming Adoption Challenges," *Journal of Systems and Software* (128), pp. 72–86.
- Cockburn, A., and Highsmith, J. 2001. "Agile Software Development, the People Factor," *Computer* (34:11), pp. 131–133.
- Dennehy, D., and Conboy, K. 2017. "Going with the Flow: An Activity Theory Analysis of Flow Techniques in Software Development," *Journal of Systems and Software* (133), pp. 160–173.
- Deshpande, A., Sharp, H., Barroca, L., and Gregory, P. 2016. *Remote Working and Collaboration in Agile Teams*.
- Dybå, T., and Dingsøyr, T. 2008. "Empirical Studies of Agile Software Development: A Systematic Review," *Information and Software Technology* (50:9–10), pp. 833–859.
- Eisenhardt, K. M. 1989. "Building Theories from Case Study Research," *Academy of Management Review* (14:4), pp. 532–550.
- Elbanna, A., and Sarker, S. 2016. "The Risks of Agile Software Development: Learning from Adopters," *IEEE Software* (33:5), pp. 72–79.
- Erich, F., Amrit, C., and Daneva, M. 2014. "Report: Devops Literature Review," *University of Twente, Tech. Rep.*
- Erich, F. M. A., Amrit, C., and Daneva, M. 2017. "A Qualitative Study of DevOps Usage in Practice," *Journal of Software: Evolution and Process* (29:6), 00:1–??
- Fitzgerald, B., Hartnett, G., and Conboy, K. 2006. "Customising Agile Methods to Software Practices at Intel Shannon," *European Journal of Information Systems* (15:2), pp. 200–213.
- Fitzgerald, B., and Stol, K.-J. 2017. "Continuous Software Engineering: A Roadmap and Agenda," *Journal of Systems and Software* (123), pp. 176–189.



- Forsgren, N., Kim, G., Humble, J., Brown, A., and Kersten, N. 2017. "The 2017 State of DevOps Report," Portland, OR, USA: Puppet and Dora, pp. 1–53.
- Ghobadi, S., and Mathiassen, L. 2016. "Perceived Barriers to Effective Knowledge Sharing in Agile Software Teams," *Information Systems Journal* (26:2), pp. 95–125.
- Hackman, J. R., and Oldham, G. R. 1980. *Work Redesign*, (1st ed.), Reading, MA, USA: Addison Wesley.
- Hemon, A., Lyonnet, B., Rowe, F., and Fitzgerald, B. 2018. "Conceptualizing the Transition From Agile to DevOps: A Maturity Model for a Smarter IS Function," in *IFIP WG 8.6 Working Conference*, Portsmouth, UK: Springer.
- Herzberg, F. 1964. "The Motivation-Hygiene Concept and Problems of Manpower.," *Personnel Administration* (27), pp. 3–7.
- Highsmith, J. 2009. *Agile Project Management: Creating Innovative Products*, (2nd edition.), Boston, MA, USA: Addison Wesley.
- Highsmith, J., and Cockburn, A. 2001. "Agile Software Development: The Business of Innovation," *Computer* (34:9), pp. 120–127.
- Humble, J., and Farley, D. 2010. *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation (Adobe Reader)*, Boston, MA, USA: Pearson Education.
- Humble, J., and Molesky, J. 2011. "Why Enterprises Must Adopt Devops to Enable Continuous Delivery," *Cutter IT Journal* (24:8), p. 6.
- Humble, J., Molesky, J., and O'Reilly, B. 2015. *Lean Enterprise: Adopting Continuous Delivery, Devops, and Lean Startup at Scale*, Sebastopol, CA, USA: O'Reilly Media.
- Karlström, D., and Runeson, P. 2005. "Combining Agile Methods with Stage-Gate Project Management," *IEEE Software* (22:3), pp. 43–49.
- Keil, M., Tan, B. C., Wei, K.-K., Saarinen, T., Tuunainen, V., and Wassenaar, A. 2000. "A Cross-Cultural Study on Escalation of Commitment Behavior in Software Projects," *MIS Quarterly*, pp. 299–325.
- Kim, G., Debois, P., Willis, J., and Humble, J. 2016. *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*, Portland, OR, USA: IT Revolution.
- Kontio, J., Høglund, M., Ryden, J., and Abrahamsson, P. 2004. "Managing Commitments and Risks: Challenges in Distributed Agile Development," in *Software Engineering, 2004. ICSE 2004. Proceedings. 26th International Conference On, IEEE*, pp. 732–733.
- Korhonen, K. 2009. "Migrating Defect Management from Waterfall to Agile Software Development in a Large-Scale Multi-Site Organization: A Case Study," in *International Conference on Agile Processes and Extreme Programming in Software Engineering*, Springer, pp. 73–82.
- Laukkanen, E., Itkonen, J., and Lassenius, C. 2017. "Problems, Causes and Solutions When Adopting Continuous Delivery—A Systematic Literature Review," *Information and Software Technology* (82), pp. 55–79.
- Locke, E. 1976. "The Nature and Causes of Job Satisfaction," *Handbook of Industrial and Organizational Psychology*, pp. 1297–1349.
- Lwakatare, L. E., Karvonen, T., Sauvola, T., Kuvaja, P., Olsson, H. H., Bosch, J., and Oivo, M. 2016. "Towards DevOps in the Embedded Systems Domain: Why Is It so Hard?," in *System Sciences (HICSS), 2016 49th Hawaii International Conference On, IEEE*, pp. 5437–5446.
- Lwakatare, L. E., Kuvaja, P., and Oivo, M. 2016. "An Exploratory Study of DevOps Extending the Dimensions of DevOps with Practices," in *Proceedings of the 11th International Conference on Software Engineering Advances (ICSEA), Rome, Italy: IARIA*, pp. 91–99.
- Mannaro, K., Melis, M., and Marchesi, M. 2004. "Empirical Analysis on the Satisfaction of IT Employees Comparing XP Practices with Other Software Development Methodologies," in *International Conference on Extreme Programming and Agile Processes in Software Engineering*, Springer, pp. 166–174.
- McKnight, D. H., Phillips, B., and Hardgrave, B. C. 2009. "Which Reduces IT Turnover Intention the Most: Workplace Characteristics or Job Characteristics?," *Information & Management* (46:3), pp. 167–174.
- Melnik, G., and Maurer, F. 2006. "Comparative Analysis of Job Satisfaction in Agile and Non-Agile Software Development Teams," in *International Conference on Extreme Programming and Agile Processes in Software Engineering*, Springer, pp. 32–42.
- Nerur, S., Mahapatra, R., and Mangalaraj, G. 2005. "Challenges of Migrating to Agile Methodologies," *Communications of the ACM* (48:5), pp. 72–78.
- Nybom, K., Smeds, J., and Porres, I. 2016. "On the Impact of Mixing Responsibilities Between Devs and Ops," in *International Conference on Agile Software Development*, Springer, pp. 131–143.

- Nyffjord, J., and Kajko-Mattsson, M. 2007. "Commonalities in Risk Management and Agile Process Models," in *Software Engineering Advances, 2007. ICSEA 2007. International Conference On*, IEEE, pp. 18–18.
- Pentland, B. T. 2003. "Sequential Variety in Work Processes," *Organization Science* (14:5), pp. 528–540.
- Petersen, K., and Wohlin, C. 2009. "A Comparison of Issues and Advantages in Agile and Incremental Development between State of the Art and an Industrial Case," *Journal of Systems and Software* (82:9), pp. 1479–1490.
- Pikkariainen, M., Haikara, J., Salo, O., Abrahamsson, P., and Still, J. 2008. "The Impact of Agile Practices on Communication in Software Development," *Empirical Software Engineering* (13:3), pp. 303–337.
- Reifer, D. 2002. "Ten Deadly Risks in Internet and Intranet Software Development," *IEEE Software* (19:2), pp. 12–14.
- Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L. E., Tiihonen, J., and Männistö, T. 2016. "DevOps Adoption Benefits and Challenges in Practice: A Case Study," in *International Conference on Product-Focused Software Process Improvement*, Springer, pp. 590–597.
- Rolland, K. H., Fitzgerald, B., Dingsøyr, T., and Stol, K.-J. 2016. "Problematizing Agile in the Large: Alternative Assumptions for Large-Scale Agile Development," in *Proceedings of the 37th ICIS International Conference on Information Systems*, Dublin, Ireland, pp. 1–21.
- Shropshire, J., Menard, P., and Sweeney, B. 2017. "Uncertainty, Personality, and Attitudes toward DevOps," in *Proceedings of the 23rd AMCIS (Americas Conference on Information Systems) - SIGADIT*, Boston, MA, USA.
- Shropshire, J., and Sweeney, B. 2017. "On Devops and Workforce Morale," in *Proceedings of the SAIS Southern Association for Information Systems*, St. Simons Island, GA, USA.
- Smeds, J., Nybom, K., and Porres, I. 2015. "DevOps: A Definition and Perceived Adoption Impediments," in *International Conference on Agile Software Development*, Springer, pp. 166–177.
- Solinski, A., and Petersen, K. 2016. "Prioritizing Agile Benefits and Limitations in Relation to Practice Usage," *Software Quality Journal* (24:2), pp. 447–482.
- Stähl, D., and Bosch, J. 2014. "Modeling Continuous Integration Practice Differences in Industry Software Development," *Journal of Systems and Software* (87), pp. 48–59.
- Sureshchandra, K., and Shrinivasavadhani, J. 2008. "Moving from Waterfall to Agile," in *Agile, 2008. AGILE'08. Conference*, IEEE, pp. 97–101.
- Tessem, B., and Maurer, F. 2007. "Job Satisfaction and Motivation in a Large Agile Team," in *International Conference on Extreme Programming and Agile Processes in Software Engineering*, Springer, pp. 54–61.
- Thatcher, J. B., Stepina, L. P., and Boyle, R. J. 2002. "Turnover of Information Technology Workers: Examining Empirically the Influence of Attitudes, Job Characteristics, and External Markets," *Journal of Management Information Systems* (19:3), pp. 231–261.
- Tripp, J. F., Riemenschneider, C., and Thatcher, J. B. 2016. "Job Satisfaction in Agile Development Teams: Agile Development as Work Redesign," *Journal of the Association for Information Systems* (17:4), p. 267.
- Venkatesh, V., Morris, M. G., Davis, G. B., and Davis, F. D. 2003. "User Acceptance of Information Technology: Toward a Unified View," *MIS Quarterly*, pp. 425–478.
- Version One. 2018. "12th Annual State of Agile Report. (April 2018)," Alpharetta, GA, USA, p. 15. (<https://explore.versionone.com/state-of-agile/versionone-12th-annual-state-of-agile-report>).
- Wiedemann, A. 2017. "A New Form of Collaboration in IT Teams-Exploring the DevOps Phenomenon," in *Proceedings of the 21st PACIS Pacific Asia Conference on Information Systems (Vol. 82)*, Langkawi, pp. 1–12.
- Yin, R. K. 2013. "Validity and Generalization in Future Case Study Evaluations," *Evaluation* (19:3), pp. 321–332.
- Zollo, M., and Winter, S. G. 2002. "Deliberate Learning and the Evolution of Dynamic Capabilities," *Organization Science* (13:3), pp. 339–351.