The link between transformational and servant leadership in DevOps-oriented organizations

The link between transformational and servant leadership in DevOps-oriented organizations

Krikor Maroukian
Microsoft
221 Kifssias Ave.
Marousi, Athens, Greece
+30 6956764566
krmaro@microsoft.com

Stephen R. Gulliver
Henley Business School
University of Reading
Reading, UK
+44 (0) 118 378 4422
s.r.gulliver@henley.ac.uk

ABSTRACT
DevOps is a set of agile and lean practices and principles in the context of software product development aiming to decrease mean time-to-market and mean time-to-recover-from-failure through a shift in organizational mindset-skillset-toolset. There is literature to suggest that adopting DevOps has been challenging in practice and that a particular leadership style is necessary to lead DevOps adoption. There are studies to suggest that DevOps leadership is mainly related to transformational leadership characteristics. In this research, a mixed methods approach is used. Initially, semi-structured interviews are conducted with 30 EMEA (Europe, Middle-East and Africa) agile and lean practitioners holding more than 10 years of practitioner experience (81%) from the private and public sectors. The contribution also includes an analysis and evaluation of a survey completed by 250 participants of which 93% works in Europe and Middle East and 76% has held previous leadership positions. By looking to recent literature we identified agile, lean and DevOps practices and principles. In addition, we identify benefits and inhibitors to DevOps adoption and its leadership. Our results suggest that deep rooted organizational culture and lack of DevOps definition clarity are usually considered impediments to DevOps adoption followed by poor communication and collaboration. Our results also show that certain DevOps adoption leadership characteristics are relevant to transformational leadership and servant leadership. The research results also indicate that the DevOps adoption leadership role is linked to certain metrics.

CCS Concepts
• Software and its engineering→Software creation and management

Keywords
DevOps adoption; practices; principles; leadership; metrics.

1. INTRODUCTION
In today's lightning-fast technology world, software is playing a much larger role in how companies compete across a broad range of industries.

As the basis of competition shifts to software, traditional organizations are finding that their current approaches to managing software are limiting their ability to respond as quickly as the business and the market transformation pace requires. Software has become pervasive in day-to-day human activities, and the world economy is now dependent on software use. This in turn has increased the importance of having software-intensive products and services that are useful, secure, and reliable consistently and constantly during operational use.

The 1990s saw the birth of pre-agile approaches such as the Rapid Unified Process [1] and agile approaches such as XP [2][3] which eventually led to Agile Software Development [43], which is characterized mainly by lightweight, flexible, adaptive processes linked to rapidly changing corporate business environments. The traditional highly structured approach to release and deployment management requires long release cycles in terms of months, which shifts focus to maintenance-only. This means that operations teams are practically focused on maintenance activities, such as bug fixes and performance enhancement. In other words, there is a lack of effort and time committed to new feature development [13]. Some of these highly structured frameworks, which regard change and release management processes, were part of earlier editions of ITIL®, the most widely accepted IT service management approach. However, the most recent edition ITIL®4 includes guidance on agility and leanness extending the spectrum of practices from IT service management to project management, business analysis, and risk management [5].

A retrospective view, of the last 20 years of software product development practices and principles, shows that a decline of XP publications has been succeeded by the gradual increase, since about 2009, in the popularity of agile and lean practices. In addition, DevOps and Continuous Integration, Continuous Deployment, Continuous Delivery are characterized as 'trending research topics', with considerable increases in popularity since 2014 [14].

A high-performing organization is characterized by adoption of DevOps practices by multiple teams and departments, high responsiveness to mean-time-to-recover from product system failure, i.e. end-user experience degradation, mean-time-to-market, change failure rate, and embedding security deep into the source code [15]. Leading DevOps practice and principle adoption has become a fundamental element to the success of DevOps teams [4][39]. However, there is still limited research to outline the leadership style, traits, competencies, and skillset accompanied with high-performing DevOps-oriented organizations. Speed in the development and delivery of new software features, provides the opportunity to respond quickly to customer needs, business opportunities, and get quick feedback

about new software features [16]. Feedback loops facilitate information that is useful to make informed decisions regarding software development efforts conducted by different stakeholders of the software product development value stream.

The purposes of our study, are to provide a better understanding of (RQ1) which leadership characteristics are required to enable DevOps practice and principle adoption, (RQ2) what are the DevOps adoption inhibitors (resistance factors) slowing down change and (RQ3) how should DevOps leadership be measured. The next section describes the three research questions in detail in its subsections correspondingly RQ1 in subsection 2.1, RQ2 in subsection 2.2 and RQ3 in subsection 2.3

2. BACKGROUND AND RELATED WORK

2.1 Leadership in DevOps adoption

The adoption of DevOps practices requires several factors to be taken into account. The most popular model among DevOps practices is known as CALMS (Culture-Automation-Lean-Monitoring-Sharing) [17], which requires a change of people's mindset, skillset, and toolset. This orientation requires gradual and minor changes in an organization's daily operations. For companies to move from structured to agile structures in software development there needs to be first, an adoption stage of agile practices and a shift to smaller cross-functional teams, and later, when a certain level of maturity is attained, DevOps practices can be adopted - such as automated system integration and continuous integration [18]. When continuous integration is in place, customers express an interest in receiving enhancements and bug fixes more frequently. Therefore, adoption of continuous delivery practices is required. The final step occurs when the organization not only releases software continuously but also develops mechanisms to conduct rapid experimentation to drive innovation.

Successful adoption of DevOps requires agile software development [19] [29]. For practitioners in the industry, there is a decline of interest in XP, and a steady increase in use of Scrum over time. Between 2006 and 2015 there was an increase in interest concerning continuous integration, however a sharp increase in DevOps adoption within the last few years [20]. This shift is most likely driven by DevOps leaders having acquired the competencies and skills that they need to master in order to contribute to the design, influence, and motivate cultural transformation, which is proven to be a critical success factor in DevOps adoption; making it a multidisciplinary topic that requires application of a mix of skills, practices, and principles [21].

The 'State of DevOps' Report, discovered a correlation between transformational leadership and organizational performance [23]. Transformational leadership comprises of four dimensions: idealized influence, inspirational motivation, intellectual stimulation, and individualized consideration. The transformational leader aims to inspire and transform followers by appealing to their ideas and emotions [41]. In addition, the State of DevOps Report conveys that DevOps leaders with a servant leadership mentality inspired better team performance [23]. In essence, the leader is serving rather than being served and therefore, creates an environment of trust, collaboration and reciprocal service which ultimately leads to higher performance [40].

In 1970, Robert Greenleaf published his first essay, entitled "The Servant As Leader", which introduced the term "servant leadership. "Servant leadership is a holistic leadership approach that engages followers in multiple dimensions such as relational, ethical, emotional, spiritual in order to empower them to grow into what they are capable of becoming" [44]. Servant leadership was developed as a theory of ethical leadership which is comprised of values such as integrity, altruism, humility, empathy, healing, personal growth, fairness and justice, empowerment, etc. [41]. Our study attempts to identify the characteristics presented by a mixed methods research design approach and how obtained results and outcomes relate to transformational and servant leadership.

2.2 Challenges in DevOps adoption

Following a decade of DevOps, there is not a definitive term or agreement amongst software practitioners and scholars at to what DevOps actually means having a plethora of differentiated published definitions [4][30][31][32][33][34]. Literature defines DevOps in numerous ways, although, the majority of descriptions specifies 'DevOps' as a term that is used to emphasize the collaboration between software development and operations. There is, however, a research and industrial need to develop a better understanding of the scope of DevOps [20]; since DevOps has been described as: a new role within a software organization [35]; a movement [32] for change in software industry [30]; a set of software development practices [4]; an agile approach [22]; and high velocity IT [5].

Cultural enablers, used to promote the adoption of DevOps practices, are leadership, focus on decision making, customer focus, engineering practices, learning and development, team recognition, innovation, guilds and performance feedback [21][36][37]. Moreover, to achieve performance gains, while adopting DevOps practices, the following are shown to be essential [38]:

- Tightened feedback loops between Development and Operations teams
- Established practices of automated performance monitoring
- Measurement of key performance metrics in Continuous Integration, Test and Operations teams
- Shared tools and performance metrics across teams.

According to the "State of DevOps Report" [23] there is an increasing inclusion of IT team members into DevOps teams from 16% of the respondents identified themselves as working in DevOps teams in 2014 to 27% in 2017. Furthermore, there are considerable challenges in DevOps practice adoption in the IT industry. DevOps adoption challenges include but are not limited to the insufficient communication, deep-seated company culture, industry constraints and feasibility, heterogeneous environments, DevOps is unclear but also evolving [13]. Moreover, a study indicated a comprehensive list of problems influencing poor cooperation between software development and operations [24]. However the most serious problems in poor software development, included operations not being involved in the requirements specifications, poor communication and information flow, unsatisfactory test environment, lack of knowledge transfer, systems being put into production before they are complete and operational routines not being established prior to deployment. Additionally, the complexity of performance engineering approaches is a barrier for wide-spread adoption by practitioners. Accordingly, performance engineering approaches must be lightweight and must smoothly integrate with existing tools in the DevOps pipeline [37].

For modern software companies speed facilitates fast and repeatable software development and delivery processes [25]. This is evident by the emergence and the growing interest of a continuous deployment paradigm in the software industry. Continuous deployment entails the capability of an organization to deliver new software features at multiple times and in the shortest time possible. DevOps is an approach that has been reported to enable the continuous deployment paradigm as it embodies a set of useful principles crucial to the development and deployment of software [26]. Practices that have posed as barriers to continuous deployment include time pressure, increased technical debt, customer unwillingness to update and conflicting goals between rapid released and achieving high reliability and test coverage. In addition, the adoption challenges that have also been identified in large scale organizations are cultural barriers, risk of disintermediation of roles, lack of DevOps education and awareness, resistance to change, silo mentality, and lack of strategic direction from senior management [36].

In general, organizations and IT professionals keep DevOps in high regard, but DevOps practice adoption is associated with challenges. These challenges can arise mainly from a combination of necessity in maintaining a legacy system, lack of senior management buy-in, managerial structure, and resistance [21]. Other points which are posed as barriers include blame-culture, communication difficulties, and delays in producing software releases [4] [30] [31] [32] [33] [34].

2.3 Metrics for DevOps adoption leadership

Metrics in traditional highly structured corporate environments produce development cycles that focus a lot on defect density of the software product; yet this is not the most effective way to measure quality in the context of software product development [6] [7]. The effect that traditional approaches have had to software development is that 'surrogation' can lead to enterprise strategy being replaced with metrics [27], with employees consciously aiming to contribute to local optima rather global corporate optima to increase flow in the value stream [8]. A set of organizational culture related motives that encourage a shift of mindset within a software product development team may include [9]:

- Team members seek information actively
- Team members are encouraged to learn through failure
- Team members share responsibility
- Team members are encouraged to collaborate cross-functionally
- Team members, enquire failure
- Team members share ideas openly.

For instance, software development teams commonly express significant differences in behavioral patterns of developers and testers when senior management first establishes a key performance metric of 'least defects in deployable code' into a production environment and then announce the downsizing of the quality assurance team thus introducing demotivators that contribute to suboptimal performance [6]. Software development should be attempting to get closer to the metrics most frequently utilized to evaluate the speed with which releases can move to production environments before performance inefficiencies start to appear [6]. Additionally, software development pipeline health is essential to maintaining high quality software. Measurement approaches in DevOps teams include but are not limited to source code version control, optimum branching strategy, static analysis, >80% code coverage, vulnerability scan, open source scan, artifact version control, auto provisioning, immutable servers, integration testing, performance testing, build deploy testing automated for every commit, automated rollback, automated change order, zero downtime release, feature toggle [12].

In addition to the aforementioned there is increased research interest in understanding how DevOps teams measure cognitive load using relative domain complexity without measuring lines of code produced, number of modules, classes, or methods [7]. This can be further analyzed in the context of flow metrics i.e. flow distribution, flow velocity, flow time, flow load, and flow efficiency [10] which represents the proportion of each flow work item being active in a given Scrum sprint. In particular, flow velocity measures features, defects, risks and technical debt in the product development flow whereas flow time resembles lead time and process time as defined in value stream maps [27]. Moreover, flow load represents active or waiting work in the value stream, and flow efficiency is the result of measuring flow load, i.e. duration of work inactivity in the value stream. Workflow can be further categorized according to the Deployment Pipeline stages [11]. At the requirements planning level new and unique work, including repetitive work, is considered for optimization purposes. Moreover, optimization requires fast feedback and a focus on end-to-end cycle time for an all-round customer feedback.

Another dimension to DevOps can be Microsoft's perception on the triage of people, process and technology while providing a strong focus on the five DevOps habits [28]:

1. Customer obsessed
2. You build it, you love it
3. Align outcomes, not outputs
4. Get clean, stay clean
5. Shift quality left and right

Lastly, measures should be useful and transparent to both leadership and engineering personnel in communicating progress and quality in a consistent format [9]

3. Research method

Having defined the agile, lean, and DevOps adoption benefits and challenges de-scribed in literature, it is crucial to determine whether these views align with industry domain practitioners.

3.1 Research design

This paper presents contextually relevant data generated from thirty (30) semi-structured interviews (see Figure 1) that were conducted between September 2018 and January 2019 with practitioners in companies working within a wide range of countries (Czech Republic, Estonia, Italy, Georgia, Greece, The Netherlands, Saudi Arabia, South Africa, UAE, UK).

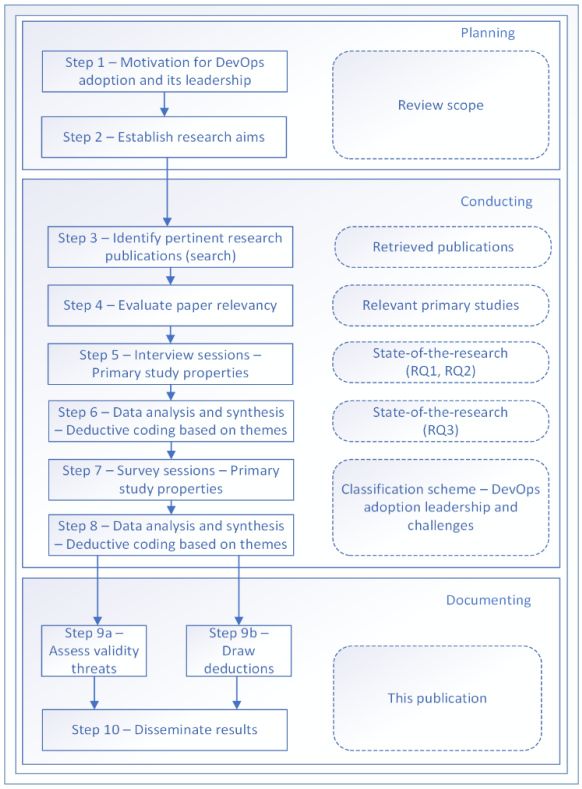

Figure 1. Research study process.

3.2 Data Collection

The interview participants were identified with their roles, organization size and country within which they work. Participants who opted for the interview process were aware of DevOps adoption practices and principles. We invited interview participants through IT events in Europe and through professional social media networks (see Table 1).To achieve a heterogeneous perspective, and to increase the wealth of information, practitioners from a variety of organizations were invited and consulted. The information provided to interview participants prior to the interview commencing stated that names or organization titles would not be disclosed as part of this research.

Table 1. Interview participant profile

P#	Job Title	PX	CN	Domain
P1	PMO Director	14	Saudi Arabia	Aviation
P2	Principal Consultant, IT Service Management	13	Italy	IT Consulting Services
P3	CIO	26	Greece	Insurance
P4	Principal Consultant, IT Service Management	11	UK	IT Consulting Services
P5	Managing Director, IT Service Management	32	UK	IT Consulting Services
P6	Smart Systems Manager	23	Greece	IT Consulting Services
P7	Senior Digital Transformation Technologist & Solution Practice Lead	30	UAE	IT Consulting Services
P8	Principal Consultant, IT Service Management	34	UK	IT Consulting Services
P9	Founding Consultant, IT Service Management	19	UK	IT Consulting Services
P10	Managing Director	29	UK	IT Consulting Services
P11	Head of Remote Transactions	16	Greece	Banking
P12	Consultant	34	Netherlands	IT Consulting Services
P13	Deputy CIO	22	Greece	Construction Management
P14	Head of Applications	18	Greece	Lottery
P15	Principal Consultant, IT Service Management	21	South Africa	IT Consulting Services
P16	Founding Consultant, IT Service Management	34	UK	IT Consulting Services
P17	Managing Director, IT Service Management	19	UK	IT Consulting Services
P18	Managing Director and Lead Consultant	14	UK	IT Consulting Services
P19	IT Operations Manager	13	Greece	Lottery
P20	IT Operations Manager	15	UK	Government
P21	Founding Consultant, IT Service Management	34	UK	IT Consulting Services
P22	Assistant General Manager, IT Operations	28	Greece	Banking
P23	CDO	13	Estonia	Government
P24	CIO	20	Greece	Insurance
P25	CIO	27	Greece	Aviation
P26	Development Team Lead	11	Greece	Lottery
P27	IT Operations Lead	12	Georgia	Government
P28	Business	18	Greece	IT Consulting

	Development Director			Services
P29	Operations and Innovation Lead, IT Services	11	Czech Republic	Courier Services
P30	CIO	28	Greece	Automotive

There were twenty (20) interview questions - consisting of two types of questions – demographic and those relative to each of the research questions – were used to shape the research survey (see Table 2).
The whole set of interview questions is available at the following URL:
https://tinyurl.com/ybxrcujq
Data collection and analysis was mapped to the research questions posed at the end of the Introduction section for interview questions (see Table 2).

Table 2. Research questions mapped to interview questions

Research Question	Interview Question
Data collection for segmentation purposes	1, 2, 3, 21
RQ1) Which leadership characteristics are required to enable DevOps practice and principle adoption?	17, 18, 19, 20, 21
RQ2) What are the DevOps adoption inhibitors (resistance factors) slowing down change?	4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 21
RQ3) How should DevOps leadership be measured?	4, 20, 21

The survey was divided into four section: 1) questions about the participant's professional information; 2) questions about DevOps practices adopted, 3) questions about leadership related to DevOps, and 4) questions on DevOps metrics. The target audience of the survey is defined mainly as Consultant, Product/Software Developer, C-Suite, Operations engineer, IT Architect. Data collection and analysis was mapped to the research questions posed at the end of the Introduction section for survey questions (see Table 3).
The whole set of survey questions is available at the following URL:
https://tinyurl.com/yapl9u3u

Table 3. Research questions mapped to survey questions

Research Question	Interview Question
Data collection for segmentation purposes	1, 2, 3, 4, 5, 6, 20
RQ1) Which leadership characteristics are required to enable DevOps practice and principle adoption?	7, 8, 9, 13, 14, 17, 18, 19, 20
RQ2) What are the DevOps adoption inhibitors (resistance factors) slowing down change?	1, 10, 12, 13, 14, 20
RQ3) How should DevOps leadership be measured?	8, 11, 15, 16, 20

The 250 participants of the survey answered six demographics questions. Additionally, the roles of the participants are Consultant (24%), Product/Software Development (18%), C-Suite (14%), Operations (11%), PMO (10%), IT Architect (8%), Business Development (5%), Information Security (4%), Director (1%), Head of Infrastructure (1%), Support (1%), IT Manager (1%), Database administration (1%), Head of Legal Department (1%), Deputy Manager IT (1%), Service Management (1%). Moreover, The industries in which the survey participants worked in are IT Services/Consulting (33%), Government (22%), Financial Services (13%), Technology/Telecommunications (8%), Manufacturing (4%), Financial Services/Consulting (3%), Aviation (3%), Construction (3%), Retail/Consumer Services (2%), Healthcare (2%), Education (2%), Recycling (1%), Insurance (1%), Energy/Utilities (1%), Lei-sure & Hospitality (1%).
A 4-point Likert scale was chosen to record opinion for the set of survey questions in an effort to add clarity to the distribution of positive or negative opinion.

4. EVALUATION AND RESULTS

The interview series consisted of thirty (30) participants from nine countries Greece (11), UK, (10), Saudi Arabia (2), Czech Republic (1), Estonia (1),Georgia (1), Italy (1), Netherlands (1),South Africa (1),UAE (1). Fifteen (15) were IT consultants and an-other fifteen (15) were from service provider organizations. The service consumers of IT consultants can be service providers or other IT consultants. The service consumers for the service provider organization can also be either internal or external. All Greek participants were service providers. UK participants consisted of nine (9) consultants and one (1) service provider. There was a distinct diversity of participant roles, e.g. Principal Consultant (9), Managing Director (4), CIO/CDO (6), IT Operations Man-ager (3), PMO Director (1), Head of Remote Transactions (1), Smart Systems Manager (1), Head of Applications (1), Development Team Lead (1), Business Development Director (1), Operations and Innovation Lead (1). Furthermore, the industries of par-ticipants were Consulting Services (14), Aviation (3), Government (3), Lottery (2), Insurance (2), Finance (2), Manufacturing (1), Logistics (1), ISV (1), Automotive (1).

The interview participants were aware of, and had considerable previous experience applying a range of frameworks, international standards, methodologies, practices, and principles; such as ITIL (26), Scrum (22), DevOps (19), Lean IT (15).

4.1 Leading DevOps

It is worth looking into the level of acceptance of a leadership role being an individual or team role and the influential effect it can

Figure 2. Coded themes generated from NVivo for thirty interview transcripts.

have on team performance in the context of software product development and coding pipeline health. Nine (9) service providers and six (6) consultants agreed that the leadership role should be an individual role whereas five (5) service providers and five (5) consultants agreed that the leadership role should be a team role. Lastly, one (1) service provider and three (3) consultants stated that both approaches are required interchangeably throughout the course of a transitioning initiative towards DevOps practice and principle adoption. The survey of 250 participants indicated that 76% agrees that a DevOps leadership role should exist.

Throughout the series of interviews there was focus on DevOps adoption and the leadership role. In fact, P5 (Managing Director, UK) and P19 (IT Operations Manager, Greece) stated that "Leadership skillset is the most important thing to adoption barrier breakdown". P7 (Consultant, UAE) stated that "In the beginning of an adoption initiative there is a constant link to fear of people losing power, loss of position, etc.". Moreover, P12 (Principal Consultant, Netherlands), P23 (CDO, Estonia), P28 (Business Development Director, Greece) and P30 (CIO, Greece) mentioned that there is "Lack of Leadership (walk-the-talk, lead by example, confront undesirable behaviors, reward new behaviors)". In addition, P23 (CDO, Estonia), P28 (Business Development Director, Greece) and P30 (CIO, Greece) added that "end-to-end ownership of the leadership role is required in terms of cross-functional team leadership". Moreover, the survey showed that 76% of participants have held or hold a leadership position and 91% claimed that the DevOps leadership role is required, and that it should be an individual role (67%). These results are similar to the results produced from the thirty (30) interview participants. The interview transcripts were imported in NVivo 12 to produce and examine the relevance of coded themes that might emerge (see Figure 2).

4.2 Challenges in DevOps Adoption

In terms of DevOps adoption inhibitors and resistance factors, P15 (Principal Consultant, South Africa) mentioned that "Extremely hierarchical organizational structures pose as a communication barrier to DevOps adoption". Another failure point for DevOps adoption can be that "DevOps practice adoption has to be at a wider enterprise scale for it to be labeled successful". In addition, P27 (IT Operations Manager, Georgia) stated that "Top management is not interested in agile and DevOps practice adoption. They do care about customer satisfaction levels which can mean a reactive attitude towards the number of complaints received". Notably, P3 (CIO, Greece) mentioned that "We identified the bottlenecks that we adopted while adopting these structured approaches". However, P8 (Principal Consultant, UK)

argues that "senior management and team members should not blame the person who introduced the new practice" since "continuous experimentation is crucial to the success of DevOps adoption and any new practice adoption". It is vital to establish the right organizational culture when it comes to the shift of mindset that DevOps adoption requires. To that extent P10 (Managing Director, UK) stated that "the team leading the adoption of the new way of working has to have the right skills and cultural drivers to succeed".

In the survey of 250 participants, 25% of respondents do not plan to adopt DevOps in the future, 30% adopted across some parts of the IT organization , 18% adopted across the IT organization, 12% adopted across the enterprise, and just 5% have not adopted nor have plans to adopt DevOps. In addition, the role in decision making process for DevOps adoption falls by 34% to C-level executive (member of Board of Directors), 21% to development team lead, 16% to product owner and 10% to architect. The high concentration of responses to C-level executive and development team lead could suggest that the development teams themselves, have to shift from a highly hierarchical organizational structure to more autonomous self-organizing team behaviors which characterize DevOps teams.

Lack of commitment by customer is recognized as the top inhibitor and resistance factor of DevOps adoption followed by a lack of organizational practice adoption capability. These results are overall expressed opinion during the interviews and indicate that there is overwhelming agreement on these type of inhibitors to DevOps adoption.

4.3 The link between Transformational and Servant Leadership in DevOps

The unison of DevOps adoption, its leadership, resistance factors and the way DevOps adoption leadership can be measured, provide insights to the leadership style that they resemble. Gaining an improved understanding of DevOps adoption leadership first requires mapping characteristics of servant and transformational leadership styles to DevOps adoption leadership, (see Table 4).

The servant leader commits time and effort to understand each follower's background, core values, beliefs and behavioral patterns not only in the professional but also in the personal domain [44]. Additionally, charismatic and transformational leaders attempt to communicate their leadership qualifications through (1) appealing to follower values (2) communicating in symbolic ways that are clear and vivid, and (3) displaying emotional conviction and passion for the mission [41]. Moreover, transformational leaders inspire and transform followers by (1) making them more aware of task outcome importance, (2) motivate to expose their self-interest for the benefit of the team or organization, and (3) activating their higher-order needs [41]. Furthermore, the primary focus in the value-based leadership styles is that a leader who has power should use that power wisely and ethically. That is the fundamental difference between value-based leadership styles and transactional leadership which is based on a relationship of value and emotions exchanged for a specific set of benefits e.g. financial gains, increased influence over subordinates, increased authoritative right [41].

There are three periods through which servant leadership has been progressing. Firstly, the period that focused on the conceptual development of servant leadership. Secondly, the period that focused on producing the measures of servant leadership and now that we are living the third period which regards the model development phase. Table 4 shows the DevOps adoption leadership characteristics identified during both; the thirty (30) interviews and the 250 participant survey.

Table 4. Characteristics of Leadership Styles

Transformational Leadership [41][43]	Servant Leadership [44]	DevOps Adoption Leadership
• Idealized influence (realistically self-confident, determined, unconventional) • Inspirational motivation (articulate, flexible, emotional, perspicacious) • Individualized consideration (caring, empathetic, relations-oriented) • Intellectual stimulation (rational, unconventional, perspicacious)	1. Empathy 2. Active Listening 3. Healing 4. Awareness 5. Persuasion 6. Conceptualization 7. Foresight 8. Stewardship 9. Commitment to the growth of people 10. Building community	1. Communication and collaboration 2. Active Listening 3. Customer-centric mindset 4. Technical background 5. Empathy 6. Multi-cultural mindset 7. Influential 8. Agile management skills 9. Strategic thinking 10. Project management skills

Lastly, similar to the general leadership field, servant leadership, transformational leadership and DevOps adoption leadership are focused on the leader-follower dyad. The dyadic relationship in all three leadership approaches can give birth to opportunities to non-traditional facets of the relationship. Additionally, another common denominator of DevOps, transformational and servant leadership styles, is the form of influence they have, based, not on tradition or formal authority but rather on follower perceptions that the leader is endowed with exceptional qualities.

4.4 DevOps Leadership Metrics

The interview series revealed that version control and issue tracking have been vastly adopted by the respondents i.e. 95%. Additionally, performance monitoring, test automation, and automated deployment seem to have important penetration in software product development practices. On the contrary, Infrastructure-as-Code, code coverage, static code analysis, trunk-based development, automated provisioning of IT resources and containerized environments didn't score as high as the aforementioned three areas. The main aim of this survey section was to uncover more around the metrics related to DevOps adoption and the leadership role. DevOps adoption practices and principles adoption levels can be measured with the traditional approach of critical success factors (CSF) (65%) and Key Performance Indicators (KPI) (63%). However, DevOps oriented metrics also gained high agreement such as mean-time-to-market (75%), deployment frequency (58%), deployment duration (53%), behavioral metrics (52%), time to detect defect (52%), mean-time-to-recover (50%). Feature usage (41%) seems to be an emerging

practice for DevOps adoption. Moreover, 91% of respondents agreed that the leadership role should be associated and have ownership of the aforementioned metrics in order to facilitate the DevOps teams efforts in the adoption of practices and principles. Lastly, regarding the software development-oriented metrics described in section 2.3, there was negligible mention in the interviews and the survey.

5. RESEARCH VALIDITY

Initially we considered the internal validity. The main validity threat relates to possible bias in the participant selection process. The communication channels, utilized to invite interview participants, were European conferences in the context of DevOps, CIO Forum and IT service management. In addition, the majority of interview participants related their work to closed-sourced software products. Concerning construct validity, there is heavy reliance on each of the interviewed practitioners' subjective perception. However, currently there is no objective approach to measure whether or not a DevOps transition journey, in the context, of practice and principle adoption within organizations can be associated to successful outcomes. The semi-structured interview series approach undertaken offers rigorous procedures for data analysis but with a certain degree of research bias. It is probable, that other researchers might deduce different findings and outcomes looking at the same set of data but the authors believe the main perceptions would be preserved. This is a typical threat related to similar studies, which do not claim to generate definitive findings. Next, we considered external validity. Although the viewpoint of the interviewed practitioners is considered with different backgrounds, working in organizations from nine (9) different industry domains and ten (10) different countries the authors do not claim that re-search results from this contribution are valid to other scenarios..

6. CONCLUSIONS

This paper indicates that transitioning to DevOps through agile and lean practice and principle adoption maintains strong linkage to the required shift of mindset-skillset-toolset. The findings are supported by thirty (30) interview participants from private and public sectors in EMEA region and the evaluation of a survey completed by 250 participants. The interviews generated coded themes to expand our understanding of relevant DevOps factors. The data collected indicates a clear list of findings that are crucial to DevOps adoption theory and is organized according to the study's research questions, as summarized below.

RQ1) What are the leadership characteristics required to enable DevOps practice and principle adoption? - From the 250 survey participants, 81% have held 10+ years of professional experience and 76% have held a leadership position. Furthermore, the survey participants indicated by 67% that a new practice and principle adoption leadership role should exist for transformation initiatives; i.e. that the C-Suite should be the direct report of the DevOps leader. The top leadership skills identified are: communication and collaboration, active listening, customer-centric mindset, technical background, problem solver, technical background, multi-cultural mindset, influential, agile management, strategic thinking, project management. Extending to the findings, the link between transformational and servant leadership in DevOps-oriented environments is shown in tabulated format. There seems to be a link at the dimensions of active listening and empathy.

RQ2) What are the DevOps adoption inhibitors (resistance factors) slowing down change? - The analysis and evaluation of interviews showed several DevOps adoption inhibitors were recognized (1) communication barriers, (2) lack of cross-functional collaboration, (3) lack of senior management buy-in, (4) lack of leadership, (5) lack of cross-functional leadership, (6) lack of enterprise-wide DevOps adoption, (7) plethora of IT systems coupled with numerous IT support roles, and (8) lack of cross-functional collaboration. In addition, the survey added (9) a lack of commitment by customer and (10) lack of organizational practice adoption capability.

RQ3) How should DevOps leadership be measured? - During the survey participants indicated that DevOps adoption leadership practices should still be governed by traditional approaches such as critical success factors (CSF) and key performance indicators (KPI). However, agile and lean metrics formed a significant part of the wider picture with the most popular being (1) mean-time-to-market (2) deployment frequency, (3) deployment duration, (4) behavioral metrics, (5) time-to-detect-defect, (6) mean-time-to-recover and (7) feature usage. The majority of respondents indicated that the DevOps leadership role should be associated and have ownership of the aforementioned metrics.

Presently, we conclude that DevOps adoption leadership is an interdisciplinary natured leadership requiring a specific set of competencies and capabilities built on a set of DevOps practices and principles. We also deduce that although there is a distinction in how DevOps adoption leadership is conceptually different from other forms of value-based leadership approaches the challenge now is to empirically test how it differs from other forms of leadership as well as cross-culturally.

7. REFERENCES

[1] Kruchten, P. 1999. *The rational unified process: an introduction*. Addison-Wesley Longman Publishing, USA.

[2] Beck, K. 2000. *Extreme programming explained: embrace change*. Addison-Wesley, Don Mills, Ontario, Canada.

[3] Fowler, M. 1999. *Refactoring: improving the design of existing code*. Addison-Wesley, Don Mills, Ontario, Canada.

[4] Bass, L., Weber, I., Zhu, L. 2015. DevOps: A Software Architect's Perspective. Addison Wesley, Massachusetts.

[5] AXELOS, 2020. *ITIL4® Managing Professional High Velocity IT*. The Stationery Office, London.

[6] Herring, M. 2018. *DevOps for the Modern Enterprise*. IT Revolution, Portland, Oregon.

[7] Kersten, M. 2018. *From Project to Product*. IT Revolution, Portland, Oregon.

[8] Goldratt, E. 1994. *Theory of Constraints and How it Should be Implemented*. North River Press, Great Barrington, Massachusetts.

[9] Herring, M., DeGrandis, D., Forsgren, N., Guckenheimer, S. 2015. *Measure efficiency, effectiveness and culture to optimize devops*. IT Revolution, Portland, Oregon.

[10] Gruver, G. 2016. *Start and Scaling DevOps in the Enterprise*. Bookbaby.

[11] Martin, K., Osterling, M. 2014. *Value stream mapping: how to visualize work and align leadership for organizational transformation*. McGraw-Hill Education, UK.

[12] Nygard, N., Pal, T., Magill, S., Guckenheimer, S., Willis, W. 2019. *DevOps Governance Architecture*. IT Revolution, Portland, Oregon.

[13] Alahyari, H., Gorschek, T., Svensson, R. 2019. An exploratory study of waste in software development organizations using agile or lean approaches: A multiple case study at 14 organizations. *Inform. Software Tech.* 105 (Jan 2019), 78-94. DOI= https://doi.org/10.1016/j.infsof.2018.08.006.

[14] Rodríguez, R., Mäntylä, M., Oivo, M., Lwakatare, L.E., Seppänen, P., Kuvaja, P. 2018. Chapter Four - Advances in Using Agile and Lean Processes for Software Development. *Advances in Computers*. 113 (May 2018), 135-224. DOI= https://doi.org/10.1016/bs.adcom.2018.03.014.

[15] Geurts, W.J.W. 2016. Faster is Better and Cheaper. *Wiley Online*. 26, 1 (Sep 2016), 1002-1015. DOI= https://doi.org/10.1002/j.2334-5837.2016.00207.x.

[16] Schlossnagle, T. 2017.Monitoring in a DevOps world. *ACM Queue*.15, 6. *Association for Computing Machinery*, New York.

[17] Willis, J. - *What DevOps means to me*, https://blog.chef.io/what-devops-means-to-me/, last accessed 2020/10/15.

[18] Rodríguez, P., Haghighatkhah, A., Lwakatare, L.E., Teppola, S., Suomalainen, T., Eskeli, J., Karvonen, T., Kuvaja, P., Verner, J.M., Oivo, M. 2017. Continuous deployment of software intensive products and services: A systematic mapping study. *J. Syst. Software*, 123 (Jan 2017), 263-291. DOI= https://doi.org/10.1016/j.jss.2015.12.015.

[19] DORA and Google Cloud. Accelerate State of DevOps Report 2019, https://services.google.com/fh/files/misc/state-of-devops-2019.pdf, last accessed 2020/10/15.

[20] Dingsøyr, T., Lassenius, C. 2016. Emerging themes in agile software development: Introduction to the special section on continuous value delivery. *Inform. Software Tech.*, 77 (Sep 2016), 56-60. DOI= https://doi.org/10.1016/j.infsof.2016.04.018.

[21] Jones, S., Noppen, J., Lettice, F. 2016. Management challenges for DevOps adoption within UK SMEs. In *Proceedings of the 2nd International Workshop on Quality-Aware DevOps* (Saarbrücken, Germany, July 21, 2016). QUDOS '16. ACM, New York, NY, 7-11. DOI = https://doi.org/10.1145/2945408.2945410

[22] Xiaofeng, W., Conboy, K., Cawley, O. 2012. "Leagile" software development: An experience report analysis of the application of lean approaches in agile software development. *J. Syst. Software*, 85, 6 (June 2012), 1287-1299. DOI= https://doi.org/10.1016/j.jss.2012.01.061.

[23] Puppet. State of DevOps Report 2019, puppet.com/resources/report/state-of-devops-report/, last accessed 2020/10/15.

[24] Iden, J., Tessem, B., Päivärinta, T. 2011. Problems in the interplay of development and IT operations in system development projects: A Delphi study of Norwegian IT experts. *Inform. Software Tech.*, 53, 4 (April 2011), 394-406. DOI= https://doi.org/10.1016/j.infsof.2010.12.002.

[25] Feitelson, D., Frachtenberg, E., Beck, K. 2013. Development and Deployment at Facebook. IEEE 17, 4 (Feb 2013) 8-17. *IEEE Internet Computing*. DOI= https://doi.org/10.1109/MIC.2013.25.

[26] Humble, J., Molesky, J. 2011. Why enterprises must adopt devops to enable continuous delivery. *Cutter Business Technology Journal*, 24, 8, 6-12. Cutter Consortium.

[27] Harvard Business Review - Don't Let Metrics Undermine Your Business, hbr.org/2019/09/dont-let-metrics-undermine-your-business/, last accessed 2020/10/15.

[28] Azure DevOps Microsoft Documentation. DevOps at Microsoft, docs.microsoft.com/en-us/azure/devops/learn/devops-at-microsoft/, last accessed 2020/10/15.

[29] Lwakatare, L.E., Kuvaja, P., Oivo, M. 2016. Relationship of DevOps to Agile, Lean and Continuous Deployment. *In Proceedings of the 17th International Conference on Product-Focused Software Process Improvement* (Trondheim, Norway, November 24, 2016). PROFES '16. Springer, Cham, Lect. Notes Comput Sc., 10027, 399-415.

[30] De França, B.B.N., Jeronimo, H., Travassos, G.H. 2016. Characterizing DevOps by hearing multiple voices. *In Proceedings of the 30th Brazilian Symposium on Software Engineering* (Maringá, Brazil, September 19-23, 2016). SBES '16. ACM, New York, NY, 53-62. DOI= https://doi.org/10.1145/2973839.2973845.

[31] Dyck, A., Penners, R., Lichter, H. 2015. Towards Definitions for Release Engineering and DevOps. *In Proceedings of the 3rd International Workshop on Release Engineering Proceedings* (Florence, Italy, May 19, 2015). IWRE'15, IEEE/ACM, 3-3. DOI= https://doi.org/10.1109/RELENG.2015.10.

[32] Lwakatare, L.E., Kuvaja, P., 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* (Rome, Italy, August 21-25, 2016), ICSEA'16, IARIA, 91–99.

[33] Smeds, J., Nybom, K., Porres, I. 2015. DevOps: A definition and perceived adoption impediments. *In Proceedings of Agile Processes in Software Engineering and Extreme Programming* (Helsinki, Finland, May 25-29, 2015), XP'15, Lect. Notes Bus Inf Processing, 212. Springer, Cham, 166-177. https://doi.org/10.1007/978-3-319-18612-2_14.

[34] Jabbari, R., bin Ali, N., Petersen, K., Tanveer, B. 2016. What is DevOps? A Systematic Mapping Study on Definitions and Practices. *In Proceedings of Scientific Workshop* (Edinburgh, Scotland, May 24-27, 2016), XP'16, Article 12, ACM, New York, NY, 1-11. DOI= https://doi.org/10.1145/2962695.2962707.

[35] Kerzazi, N., Adams, B. 2016. Who needs release and devops engineers, and why?, *In Proceedings of International Workshop on Continuous Software Evolution and Delivery* (Austin, USA, May 14-15, 2016), CSED'16, ACM, New York, NY, 77-83. DOI= https://doi.org/10.1145/2896941.2896957.

[36] Kamuto, M.B., Langerman, J.J. 2017. Factors inhibiting the adoption of DevOps in large organisations: South African context. *In Proceedings of the 2nd IEEE International Conference on Recent Trends in Electronics* (Bangalore, India, May 19-20, 2017), RTEICT'17, Lect. Notes Comput Sc., 48-51. DOI= http://doi.org/10.1109/RTEICT.2017.8256556.

[37] Bezemer, C.P., Eismann, S., Ferme, V., Grohmann, J., Heinrich, R., Jamshidi, P., Shang, W., van Hoorn, A.,

Villavicencio, M., Walter, J., Willnecker, F. 2019. How is Performance Addressed in DevOps?, *In Proceedings of the International Conference on Performance Engineering* (Mumbai, India, April, 2019), ICPE'19, ACM/SPEC, New York, 45-50. DOI= https://doi.org/10.1145/3297663.3309672.

[38] Gottesheim, W. 2015. Challenges, benefits and best practices of performance focused DevOps. *In Proceedings of the 4th International Workshop on Large-Scale Testing Proceedings*, (Austin, USA, February 1, 2015). DOI= https://doi.org/10.1145/2693182.2693187.

[39] Maroukian, K., Gulliver, S.R.: Leading DevOps practice and principle adoption. *In Proceedings of the 9th International Conference on Information Technology Convergence and Services* (Vancouver, Canada, May 30-31, 2020), ITCSE'20. 10, 5, AIRCC, Computer Science and Information Technology, 41-56.

[40] Greenleaf, R.K. 2002. Servant leadership: A journey into the nature of legitimate power and greatness. Paulist Press.

[41] Yukl, G.A., Gardner, W.L. III. 2020. Leadership in organizations. Pearson Education, Essex.

[42] Poppendieck, M., Poppendieck, T. 2003. Lean Software Development: An Agile Toolkit. Addison-Wesley Professional, Boston.

[43] Bass, B.N., Riggio, R.E. 2006. Transformational Leadership. Lawrence Erlbaum Associates, New York.

[44] Eva, N., Robin, M., Sendjaya, S., Dierendonck, D.V., Liden, R.C., 2019. Servant Leadership: A systematic review and call for future research. The Leadership Quarterly, 30, 1, 111-132. DOI= https://doi.org/10.1016/j.leaqua.2018.07.004

Authors' background

Your Name	Title*	Research Field	Personal website
Krikor Maroukian	**PhD Candidate (Henley Business School, UK)**	**Agile, Lean and DevOps in Software Product Development**	**www.linkedin.com/in/krikormaroukian/**
Stephen R. Gulliver	**Associate Professor**	**Pervasive Informatics**	**https://www.henley.ac.uk/people/person/dr-stephen-gulliver**