Staqo
Let's Simplify

White Paper

DevOps for Enterprises: Bringing IT and Operations Together

The Compelling Shift for Agile Enterprise Applications – Navigate It With The Right Tools

Authors:
Narin Bhandari
Pertisth Mankotia

Foreword

In an age where an enterprise's business agility is fuelled by the speed, performance, and robustness of its applications, DevOps was an inevitable shift. The gravity and speed at which it took the center-stage spot testifies to the importance that application agility commands in the digital world. Development teams and operations teams have, for long, suffered the consequences and burdens of working in isolation. They have seen a lot of time, impact, room for course-correction and insights wasted in translation loss. DevOps has crushed those boundaries and has ushered in unprecedented levels of efficiencies and synergies.

But not every enterprise that flicks the switch to DevOps can reap its gains from the very first week. There are tools to be embraced, mastered and applied well to get the execution part of DevOps right. This whitepaper delineates some salient areas of DevOps challenges and guides on adoption of paths that lead to the intended outcomes. In fact, it is quite possible to find a balance between both Dev and Ops once an enterprise has identified the unique value it can bring to the business.

Contents

Staqo
Let's Simplify

Deconstructing DevOps

The Need

DevOps is the new way of IT life. It has finally broken the walls between two rigid (but equally important) force-fields of any enterprise application – Dev and Ops. As a set of tools, practices, and cultural changes, DevOps has shaken the erstwhile way in which applications were made.

It essentially brings the development teams and operations teams on the same page, breaking years of infrastructure and thought silos. The idea is to design, develop, test, and manage applications with the integration of 'Dev' and Ops' at the right places. Before DevOps, the two sets had different constraints, different goals, different functions and different floors that they worked on. But DevOps put them together in the same room – metaphorically. Now those who write code do not have to be miles away (organizationally and functionally) from those who deploy and support that code.

> DevOps is bringing development and operations teams on the same page, breaking years of infrastructure and thought silos.

The Gains

Here's what happens when DevOps kicks in:

- A holistic look at the development process where needs and perspectives of all dimensions are taken care of
- Improved agility, swifter incident response time
- Lower operating costs due to removal of disparate tools and processes
- Stoppage of context-switching
- Development pipeline stays unblocked, clear and customer-oriented
- Continuous delivery and shift-left controls
- Enhanced speed
- Higher levels of quality and debugging
- Comprehensive and early testing in production-like environments
- Error reduction
- Early access to source code for developers for debugging
- Mitigated risks of deployment
- Removal of unnecessary code
- Scalability
- Shrinkage in dependencies
- Ease for cloud-native and mobile applications
- Readiness for multi-cloud context and application modernization
- Faster movement from systems-of-record fields to new cloud models and systems-of-engagement
- Quick management of failures, fast recoveries
- Fast and smooth project paths
- Stronger application resilience
- Better time-to-market windows
- Amplified customer experiences
- A new culture of collaboration, transparency, systems thinking, shared responsibility, and joint problem-solving

According to an EMA research summary 2019, when assessed on a quantitative scale between seamless and confrontational, DevOps interactions were often found to be positive (46 percent were seamless and only 15 percent were leaning toward confrontational). Also, the correlation between DevOps and digital transformation was found to be quite significant. These findings corroborate the cross-sectional thumbs-up that has been seen in the IT landscape for DevOps. Industries of all types and legacy-ages have shown interest, if not investments, in this big shift in the way applications are thought, built, and delivered. Only 12 percent of respondents in the EMA survey were seen with no DevOps/ITSM handshake. Incidentally or not, when compared with the other groups, they were consistently the least effective.

That explains why over half of custom enterprise applications are using agile methods as of 2019, with 81 percent finding that their production deployments have increased. The net effect of DevOps on operations has been reckoned to be positive for 55 percent of respondents, and negative for only 10 percent respondents.

Perhaps, it is because DevOps automates some key areas and injects the agile approach for operational efficiency and standardization. With DevOps in place, enterprises can release applications with a never-before speed, confidence, and efficiency.

It helps the teams and users to say goodbye to monolithic and complicated applications that had a lot of deadweight. It got rid of that huge translation loss between what the Ops wants and what the Dev makes.

The adoption and appetite for DevOps has found timely catalysts. As organizations push the pedal on cloud, digital transformation, and on the use of agile frameworks, they need something that can wipe away the inefficiencies faced across the software development lifecycle. The result is a new environment of continuous integration, testing, automated software development, and delivery of software. It automates the movement and deployment of code across different environments. It shapes a continuous feedback loop so that development teams can shorten response time and assure continuous release of software.

The Growth Frontiers

As per a report from MarketsandMarkets, the DevOps market is expected to grow from USD 3.42 billion in 2018 to USD 10.31 billion by 2023, at a Compound Annual Growth Rate (CAGR) of 24.7 percent. North America was slated with the largest market size in 2018, but the Asia Pacific (APAC) region was projected to grow at the highest CAGR during the forecast period in the DevOps market. Among the many drivers spotted, the top ones were: Need for efficient computing framework and complete security while operating in physical, virtual, or cloud environments are expected to drive the global market.

Another report by Global Market Insights, Inc tells that this market will reach USD 17 billion by 2026. The Operations DevOps market is expected to show around 22 percent gains from 2020 to 2026. The cloud-based deployment model is slated to grow exponentially in the DevOps market over the forecast timeline with a CAGR of 20 percent. If we see a study done by Grand View Research, Inc the global market could be valued at about USD 12.85 billion by 2025. This is being propelled by rapid adoption of digital services, advanced cloud infrastructure, and presence of major technology players.

It is the perfect world for applications with maximum business impact, lower costs, great flexibility, impressive agility, and fast application delivery.

Challenges that interrupt the 'Match-Making'

Not all enterprises find DevOps to be a walk in the park. Many issues crop up and have to be addressed:

- DevOps can lead to changes to an organization culture that slows down project delivery
- IT Complexity can get out of control
- Dev gets faster with DevOps but Ops is put on a back-burner
- App quality improvements do not align with DevOps fast-tracks
- Business stakeholders are not patient or ready enough
- Unrealistic expectations
- ITSM misalignment
- Insufficient automated test coverage and preponderance of manual testing processes
- Inadequate emphasis on architecture and design
- Continued siloed troubleshooting instead of an integrated approach
- Limited tool support across the lifecycle
- Lack of confidence in measurement of DevOps
- Unexpected, sudden, and high costs of implementation
- Confusion and uncertainty around SLAs and Resources

In the EMA survey, the most critical technology gaps across the DevOps divide were as follows: Application Performance Monitoring (APM) (four percent); End-user experience analytics (26 percent); Network management (24 percent); Collaboration capabilities (24 percent); and ITSM tools (22 percent). What is really worth noting is that DevOps solutions and teams are turning out to be less effective in areas like addressing project backlogs in operations (50 percent), support for hybrid environments (40 percent), migrations to private cloud (39 percent), transition from product dev and production (35 percent), migrations to public cloud (32 percent), lifecycle-level process-orchestration (22 percent), migration to micro-services and containers (21 percent), support of pre-production requirements (eight percent) and correlation of services to business outcomes (four percent).

In the 2020 DevOps Trends Survey by Atlassian & CITE Research report, we also saw that DevOps practitioners (62 percent against 49 percent decision-makers) find it difficult to measure the impact of DevOps progress and success. It is not the decision but the execution which is, apparently, manifesting into on-ground problems when DevOps starts rolling. Decision-maker employees have been found to be more likely to expect a positive impact from DevOps (75 percent compared to only 60 percent of practitioners). Wrong and insufficient tools have been looked upon as a key barrier in DevOps implementation (24 percent) and lack of a clear plan (12 percent) has also surfaced as a roadblock here.

What really catches one's eyes though is this insight - organizations are statistically less likely since 2018 to be measuring customer satisfaction on new releases now. That cannot be the way to hide one's head in the sand. There are more ways than playing Ostrich to fix DevOps before it's too late.

Tools - What it all ultimately boils down to

Too many tools. Fragile tools. Fragmented tools. No matter how/what we call it, the underlying problem cannot be wished away. Tools define both good and bad DevOps, depending on how well an enterprise chooses and deploys them.

Tool fatigue can end up ruining the most solid DevOps environments. If the teams have to spend too much time to understand, connect, and apply a huge pile of tools, it ends up defeating the promise of speed and ease DevOps stands for. To that end, the feedback systems should be instant and available at the right spots to ensure that DevOps delivers fully.

Hence, it is important to:

- Choose tools with high code maintainability, good configuration and integration ease, scalability, along with readiness for cloud, containers, and AIOps
- Stress on tools that enable early diagnosis of errors and are easy on other areas too – such as upgrades, self-service, testing, compile time, costs, CI-CD, automation, container orchestration
- Focus on versatility – Tools with a wide range of plug-ins and pre-built features can deliver better results
- Go for low-intensity, automated, data-driven and business-impact testing at the right places
- Plan for skills - As per DevOps Institute research, 55 percent of survey respondents prefer to hire into their DevOps teams from within their organization. Unfortunately, many companies struggle with the availability of DevOps skills, and most cannot add or stretch these competencies due to budget restraints.
- Remember that tools should decrease and not add to technical debt. Avoid tools that do so through scripts, configuration files, or known bugs that go unfixed in favor of new features, insufficient test coverage, code or artifacts that aren't cleaned up when no longer used. Avoid tools with sloppy documentation, incomplete migration, issues in infrastructure, as well as application code. Remember, those with high technical debt have been found to be 1.6 times less productive, and the highest performers have been seen as 1.4 times more likely to have low technical debt.
- Resist the temptation - Do not choose tools based only on how widely used or affordable they are. Think of the hands that will drive them. Think of power-users like UX, infosec, and database engineers. This is important because a high level of correlation has been found in usefulness and ease-of-use of this deployment tooling with CI and CD outcomes and productivity.
- Focus on Software Delivery and Operational Performance (SDO): In a research from DevOps Research and Assessment (DORA), we can see five areas that teams must improve in 2020 - Lead Time, Change Fail, Availability, Deployment Frequency, and Time-to-restore Metrics. Elite performers reported a change failure rate between zero and 15 percent, but low performers reported change failure rates of 46 percent to 60 percent.

Hallmarks of the Right DevOps Strategy

For those organizations that are investing in Cloud and Containers, it is essential to note that a successful DevOps scenario is deeply intertwined with the yields expected from Cloud IT. The core backbone of any Cloud-driven transformation hinges strongly on the automated provisioning, rapid development; and continuous quality and integration that DevOps makes possible.

Notably, as per the latest DORA check - the 2019 State of DevOps Report, delivering software quickly, reliably, and safely is pivotal to technology transformation and organizational performance. It has also been clearly reinforced that the best strategies for scaling DevOps in organizations focus on structural solutions that build community. As to the section of elite performers – Cloud continues to stand out to drive high performance. The best teams are seen recovering from incidents 2.604x faster - and with 7x lower change fail rates. The best teams are also deploying 208x more frequently and have lead times 106x faster when compared to low performers. Is that you? Can that be you? How?

Consider this checklist to know if your DevOps is on the intended tracks here:

- Instant and unlimited provisioning of resources
- Rapid build, testing and deployment capabilities
- Enablement of diverse needs and platforms of any Cloud scenario
- Smooth jump from the software-maker's approach to a services-delivery mindset
- Shift of focus from features to application outcomes
- Deep dialogue between IT and Business
- Proactive work on application requirements, design and testing
- Heightened hardware independence and platform-agnostic strengths
- Ability to live up to the scale and intensity of production-level workloads
- Overlay teams between Dev and Ops initiatives
- Common data reference for Dev and Ops on eventual customer experience data
- Higher quality deliverables
- Faster deployment frequency as well as recovery cycles
- Lower failure rate of new releases
- Application prioritization and clarity
- Executive-buy-in at senior level
- Good results on metrics like application crash rates, mean time recovery, team velocity, feature usage, and mean time between failures
- Press-button ease of application development and on-ramping on Cloud
- Readiness for new advancements like NoOps, NoCode, SecOps and AIOps
- Near real-time customer responsiveness that is manifested across the application cycle, specially for quality control and debugging red-flags
- Well-mapped out roadmap where costs, resources, tools and milestones align without abrupt conflicts
- Smooth and profitable translation of DevOps goals into execution areas

Conclusion and Tips

Do not forget that the fruits of DevOps are not the low-hanging ones. They need some time and effort before they can be grabbed as intended. This is where you would need to invest in:

- Organization-specific 30,000 feet planning
- Investment in culture-reorientation
- Availability of skills, learning and training in the new environment
- Assessment of tools on a wide range of metrics
- Long-term horizon for calculation of impact and cost-analysis

DevOps cannot replace the gap of a right product strategy. It is futile to release software faster, if it is not good enough in the first place for a customer or market need. In what Gartner released as a Product Management Framework in 2019, we find 22 areas to reduce risk and maximize results. Turns out that DevOps has a direct relationship with one single area: Accelerate time-to-market. But poor development process or bad design cannot be areas where DevOps will work like a silver bullet.

When DevOps is done well, the litmus test is clear - Dev does not depend on Ops or vice-versa – both act in concert for excellent customer experiences as the core goal of application development.

Overall, in a condensed way, the answer is simple. Start working with experts who know DevOps in and out. A smart enterprise spends as much time on 'how' to execute a DevOps journey as it does on 'when' to start the DevOps journey. While the initial phases can be intimidating and overwhelming, a ruthless focus on low cost and low risk can manifest actual gains for the enterprise while helping every stakeholder in the process. This is where the choice of the right DevOps partner gets more acute than ever. A good team with experience and best-practices on a variety of scenarios can help an enterprise wiggle out of even a messy situation that might arise on the course of DevOps deployment.

In no instance you should be spending unnecessarily – time, money or people- on any part of DevOps. Experts can guide you on how exactly to avoid these bumps – by use of open source tools, a balanced kit, apt licensed apps and the right SLA approach.

So ensure that you use partners that bring a versatile portfolio experience and a rigorous outcome-oriented approach. This will ensure that DevOps yields the amplified customer impact that an enterprise aspires to accomplish. Devops is not complicated. Do not let the tools (you use) make it so. Un-complicate it with the right approach. Make it compelling.

Staqo
Let's Simplify

For more information on how Staqo helps organizations leverage DevOps to enhance their operational & IT agility, please reach us at **contactus@staqo.com**

www.staqo.com

Staqo
Let's Simplify