DevOps in the Public Sector

A ReleaseTEAM Press Publication



RT RELEASETEAM
DevOps Tool Specialists

Contents

Why DevOps and How to Pitch It to the Bureaucracy
DevOps as a Government Agency Culture
The Benefits of Adopting a DevOps Strategy Within a Government Agency
Moving to a DevOps Lifecycle Model Within a Public Entity

Adopting Agile Methodologies Within a Bureaucracy **4**
The Challenges with Adopting Agile in a Bureaucratic Institution
Finding a Balance Between Agility and Oversight

DevOps and Your Legacy Systems **5**
Legacy Systems and Technical Debt
DevOps Solutions in a Legacy System
DevOps Technologies that Enable Legacy Migration

Securing Government DevOps Workflows **6**
The Security Challenges Facing Public Entities with DevOps
DevSecOps Leverages Automation to Ensure Security
On-Site Deployments of DevOps Solutions

Essentials for Completing a DevOps Implementation Project Within Budget **8**
The Steps and Benefits of DevOps Implementation in an Organization
Budgeting for the Essentials in the DevOps Implementation Project
Resource Allocation
Technology Selection
Setting up the Environment
Training Staff

Staffing Your DevOps Project or Team on a Budget **9**
The Challenges with In-House DevOps Implementations
The Cost of Getting DevOps Right from the Start
The Zero to DevOps Hero Hiring Approach

Conclusion **10**

ReleaseTEAM's Service and Solution Offerings **11**
Software Sales and Support Services
Staff Training and Mentoring
GSA and Development Outsourcing
About ReleaseTEAM, Contact Info

The DevOps revolution continues to add value to private entities, helping companies build and release software quickly and efficiently.

Traditionally, a new software development model isn't something the public sector pays a lot of attention to, as a trial-and-error approach to update processes and tools carries significant risks. However, with a DevOps strategy, this isn't the case. Government agencies and public entities are actively embracing the philosophy, even coining the moniker "Federal DevOps" for the new discipline.

The DevOps Application Lifecycle Management (ALM) model is no longer a fringe set of technologies, processes, and automations. Over the last decade, DevOps has become a comprehensive set of strategies that improve the public sector's build stability, release frequency, team collaboration, as well as end-user satisfaction, and system security.

Why DevOps and How to Pitch It to the Bureaucracy

The agility that comes from a DevOps workflow is what makes it so attractive to modern companies. Similarly, with government systems and services, the public expects the same standard of delivery that technology-first organizations provide. If your agency implements DevOps processes correctly, you can expect greater processing accuracy without compromising compliance with the regulatory frameworks involved.

Many challenges exist within the public sector that doesn't necessarily hamper private companies in the same way. Funding new initiatives and ensuring changes to the process doesn't expose agencies to data-security vulnerabilities are some of the primary risks among them. The flexibility and improved oversight that the DevOps Lifecycle provides can easily overcome these challenges. When approaching the departmental heads, it's important to highlight that DevOps is a culture that establishes Continuous Integration and Continuous Delivery (CI/CD) pipelines.

DevOps as a Government Agency Culture

Every pipeline requires a team responsible for the build, test, package, release, and monitoring of every update performed in the system. With shared responsibilities, the team needs to collaborate extensively, automate as much of the processes as possible, and build effective closeout and feedback loops for every issue or feature request solved. Security needs to provide input to the teams early and ensure every release complies with the related information security policies. Naturally, this culture of continuous development and collaboration should start with support from the top. If the agency and department heads do not buy into this culture, the adoption would likely not succeed.

The Benefits of Adopting a DevOps Strategy Within a Government Agency Include:

- Improved oversight of the entire lifecycle from requirement solicitation to deployment and system monitoring.
- Reduced resource overhead with automation and standardization tools.
- Demonstrable compliance with internal and external information management policies.
- Continuous improvement of code reliability while speeding up the cycles of new feature releases.
- A more satisfied and engaged audience

With a DevOps culture, agencies can modernize legacy technologies using microservice architecture, containerized application deployment, and serverless programming strategies. Part of the process requires centralizing data and communications, leading to the entire agency benefitting and not just the technology or IT departments that implemented the methodology.

Moving to a DevOps Lifecycle Model Within a Public Entity

For agencies who are looking to improve on their delivery and modernize or digitize services, engaging with a solution provider and experienced DevOps consultancy remains the best option. This reduces the training and skills transfer overheads, provides access to real-world expertise, and ensures the adoption of best practices from the project's outset.

Adopting Agile Methodologies Within a Bureaucracy

While the private sector was quick to embrace agile development methodologies, the necessity for bureaucratic oversight hampered adoption from public entities. Finding a balance between the structured trust built into the waterfall model while improving release cycles remains a complex task.

Most developers working in public entities or government agencies will be the first to tell you that any new methodology (such as a DevOps workflow) sounds like science fiction when it comes to the daily tasks and activities performed at these institutions.

When bureaucracies first emerged, people considered it an innovative idea that helped companies and organizations scale without compromising operations. The business cycles were annual, and anything relating to change would fall into that schedule.

In the current modern technological climate, a year may as well be a lifetime for business units and public stakeholders. Bugs and defects need immediate solutions to prevent massive amounts of hours spent on rework or finding ways to work around the known issues.

The Challenges with Adopting Agile in a Bureaucratic Institution

Challenges in adopting agile methodologies exist in every company, but in government and public sector organizations, the need for oversight and control compounds the issue. Firstly, departments and Business Units (BUs) have specific mandates that create competing priorities. It also creates a culture of distrust between teams, leading to increased control measures, further slowing down the software development lifecycles.

Every BU has new requirements, and often the department head who bangs on the table the loudest will get the highest priority. While not ideal, this is precisely the kind of culture that the waterfall approach created within these bureaucracies.

Some primary challenges in adopting DevOps workflows within a bureaucracy include:

- Bloated processes that still rely on physical approvals for any change.
- A cultural legacy of authority that influences the decisions relating to new initiatives.
- Budget limits (or reductions) making upgrades or implementations more difficult.
- Limited access to the latest expertise with new technology and/or tools.

While this list is not exhaustive (or unique to the public sector), the ability to overcome them remains a priority for government agencies. Modern business environments are increasingly volatile and dynamic, requiring frequent adaptions and rapid changes to solve technically complex issues. A necessary imperative exists to find an equilibrium between oversight and reacting to changes in all organizations.

Finding a Balance Between Agility and Oversight

Even within these constraints, the opportunity exists to find a balanced approach that improves code reliability and software quality. Using a sand-boxed, iterative approach will assist agencies in speeding up release cycles. Planning releases and assigning tasks within a set window of time will immediately improve the team's productivity.

Additionally, a new concept that's gaining ground is Minimum Viable Bureaucracy. Like the Minimum Viable Product strategy, this concept focuses on assigning autonomy to the responsible parties and delegating authority to help the teams achieve the desired outcomes.

Finally, getting buy-in from the top level will be beneficial but is rarely achievable within a bureaucracy. For advocates of an agile Application Lifecycle Management (ALM) model, it may be necessary to show rather than tell upper management about the benefits that methodologies such as Scrum and DevOps can bring to the team. A pilot project that demonstrates the improved collaboration and communication, as well as auditable change history, may go a long way to getting department heads onboard.

DevOps and Your Legacy Systems

Legacy systems encompass more than just the bare metal servers and software used in your organization. It could also include the physical workflows, information routing paths, domain authorities, and resource responsibilities currently implemented.

The existing system architecture would undoubtedly impede growth in the modern technology economy. Legacy systems served the old development methodologies and operational processes well. However, the longer these systems remain within an organization instead of adopting a DevOps strategy, the less efficient the teams will likely become.

Legacy Systems and Technical Debt

In legacy environments, the teams operate in silos and rarely share responsibilities. This could lead to increased technical debt over time, incurred either knowingly or unintentionally. If you don't address the technical debt, eventually, your backlog will increase, and an ever-greater amount of resources will be required to clear these tasks. Additionally, as the legacy system limits collaboration and cross-functional teamwork, rework becomes a daily activity for employees.

Clearly, for all but some completely insulated organizations or entities, legacy systems are no longer sustainable. Adopting a DevOps workflow can alleviate the dependency your organization has on the legacy environment. DevOps solves many issues that have long plagued the software development and system operation disciplines. The DevOps lifecycle relies primarily on an organizational culture that utilizes integration tools, allowing teams to take complete responsibility for development tasks and the system's daily stability.

DevOps Solutions in a Legacy System

With DevOps tools, teams branch a portion of the code repository, update the branch with fixes or features, then merge the branch back to the master daily. The deployment model allows for Continuous Integration and Continuous Delivery (CI/CD) pipelines that improve the code's reliability and enables teams to release updates with confidence more frequently.

The underlying continuous development tools and automated workflows increase communication and collaboration within departments and teams. The tools also centralize data and reduce the need for physical paper processing while retaining oversight and control of the processes. DevOps solutions provide completely integrated environments that enable legacy migration projects with improved ROI.

DevOps Technologies that Enable Legacy Migration

Using technology such as containerization, microservices, or serverless code, organizations can migrate legacy applications to a DevOps lifecycle without making drastic changes to the code repositories. The tools that manage the branching and updates to the repositories also integrate with the workflow and project management solutions, improving oversight and collaboration. By overlaying the modern technology stack onto the legacy system, teams gain efficiency and productivity, improve system reliability, and continuously reduce infrastructure overhead.

The tools that can assist with legacy migrations include:

- Atlassian Confluence – Provides centralized information and knowledge management in a digital environment.
- GitLab – A complete, end-to-end CI/CD pipeline solution for every stage in the DevOps lifecycle.
- JFrog – A suite of products that enables DevOps acceleration using artifact management principles and universal distribution systems, and automated security audits.

ReleaseTEAM's expertise in all of the above tools and processes will ensure your organization can leverage DevOps' benefits for a more significant ROI. The ability to merge legacy systems with new technology stacks will enable the organization to scale quickly without compromising system integrity.

Securing Government DevOps Workflows

The Security Challenges Facing Public Entities with DevOps

Government agencies require enhanced security due to the nature and sensitivity of the information processed daily. The DevOps lifecycle doesn't necessarily invoke a sense of information safety, simply due to the frequency of releases involved when adopting the framework. As each update presents a risk, ensuring the DevOps workflow for each build, test, stage, release, and monitoring phase remains secured is of the utmost importance.

In recent times, more and more threats expose agencies to the possibility of a data breach, leading to untold amounts of damage and even resulting in financial losses. With the recent exploits succeeding at municipal districts (where bad actors used ransomware to shut down and encrypt systems), the threat that any vulnerability poses is very real. For public entities looking to adopt a DevOps framework, knowing that the systems remain secure with every update remains a primary concern.

Every modern platform and digital service faces challenges in ensuring information remains secured while making changes to the code base. With distributed servers and development teams, centralized security becomes an additional overhead but remains a critical necessity for organizations. In this aspect, a DevOps workflow is no different from other development methodologies.

The primary DevOps security challenges include:

- Distributed credentials for different tools and platforms that manage the codebase.
- Encryption and API keys scattered throughout applications and environments and not managed centrally.
- Hard-coded credentials and keys within applications used to speed up development cycles, effectively bypassing security.
- Open source solutions that contain known vulnerabilities used in the DevOps cycles.

Traditionally, once developers commit new code into the testing environment, security engineers would carry out extensive checks to ensure none of those mentioned above vulnerabilities ended up in the production environment. However, humans do make mistakes, and developers take shortcuts during the build stages. With the advent of DevSecOps, public entities can reduce (if not eliminate) most of these risks.

DevSecOps Leverages Automation to Ensure Security

DevOps uses advanced automations to build, test, and deploy new software. The tools used for these tasks also enable security teams to automate checks and find known vulnerabilities within each new code commit. Modern DevOps solutions such as Sonatype's Nexus suite, JFrog's Xray application, Atlassian's CI/CD suite, and the GitLab platform all come with automations that include the security team's scripts and audits during the test and deploy stages.

By including the entire operational scope within the technology stack, the security team becomes a vital part of the DevOps life cycle. Using this approach, government agencies can utilize the DevSecOps model to address all the concerns related to every code build and release cycle. In fact, the DevSecOps approach ensures better collaboration and speedier execution of the updates within the teams.

On-Site Deployments of DevOps Solutions

While DevOps is synonymous with the cloud, the technology stacks are available for on-premises deployments to enhance security and oversight. For government agencies, physical controls over the infrastructure remain a priority. Although modern cloud environments that host public systems are safe (as the latest breaches occurred due to micro-configuration errors), ensuring every facet of the DevOps environment remains under the agency's control is still possible.

Essentials for Completing a DevOps Project Within Budget

Many pitfalls exist when moving to a new development strategy and software delivery methodology. While a DevOps workflow does present the most significant ROI for organizations, any adoption of such an initiative requires detailed oversight and expert planning to ensure the project's successful execution.

Specifically, traditional funding and monitoring activities may not suffice for a DevOps implementation. As the philosophy is radically different from other development models, the way public sector agencies allocate and track costs relating to the project's performance may need to be adapted accordingly.

The Steps and Benefits of DevOps Implementation in an Organization

The good news is once you've decided to implement a DevOps strategy in your organization or agency, you've succeeded in successfully advocating for change. To guarantee the project's success, you'll want to follow a formalized DevOps lifecycle for the entire implementation. The organization should assemble a team who'll take responsibility for the project, as well as understand every step and the costs associated with each element.

The project's lifecycle steps start with a detailed assessment. Gaining insights on the exact requirements will assist with determining the implementation plan. Once you've documented the existing technology, processes, and workflows, you'll be able to map out new tools and workflows for your Continuous Integration and Continuous Delivery (CI/CD) pipelines.

The next phase would be to implement a proof of concept, and once the technology achieves the stated goals, the team can build the environment. Finally, a cutover plan will help the team manage the organization-wide launch, then train and monitor the new environment before starting the closeout phase.

Budgeting for the Essentials in the DevOps Implementation Project

Quantifying the abstract steps into actual budget line items may seem daunting. Understanding that each phase will affect the project's funding in different ways will help you establish the entire project's costing.

Resource Allocation

Allocating resources with time weightings to the project starts in the assessment phase. Depending on the size and scope, the team may need to be dedicated to this initiative entirely. As the organization's operations will need to continue uninterrupted, it may be necessary to augment staff with external experts to leverage DevOps' benefits. Bringing in a single consultant early to assist with planning the entire project will ensure the team clearly understands each element required.

Technology Selection

Once again, the scope will inform the team about which tools will best serve the organization's needs. Full-stack solutions such as Atlassian can provide a comprehensive suite of tools and include project management integrations like Tempo. During the proof of concept step, the team will test different tools and workflows and establish most of the required automations. DevOps lifecycle solutions come with flexible licensing options, meaning the team can make use of trial versions or even select open-source tools for the project before deciding on the final technology stack.

Setting up the Environment

DevOps solutions leverage many technologies to reduce the infrastructure overhead required by the development and operations teams. These include containerized applications, serverless code, and microservices. The new DevOps environment will most likely reduce the necessary server and resource provisioning overhead but will need to operate concurrently with the existing environment until the cutover. Ensuring the team understands the costs associated with the environment early will help keep the project within budget.

Training Staff

It's essential to educate staff throughout the organization on the new processes and tools deployed. The organization may opt to let staff learn on the fly; however, building the training requirements into the planning stage will ensure a smooth closeout of the project. It will also prevent project costs ballooning if employees make a mistake that requires experts to rework and retrain them on the new solutions.

Staffing Your DevOps Project or Team on a Budget

One of the many complexities involved with adopting a DevOps strategy is finding the right development operations resources. As the philosophy merges the intricacies of the build, test, stage, release, and monitoring tasks by using powerful automations in the Software Development Life Cycle (SDLC), experienced resources are hard to find. If your agency or organization faces budgetary constraints (as most do), this becomes more difficult as dedicated resources don't come cheap.

For most organizations, deciding when to source a DevOps engineer is the primary concern. Usually, team leads will divvy up related tasks at the start of the project among the existing developers and IT operations departments. As the product and processes evolve to where operational requirements take up too much time, the company will look to find a dedicated individual. While this makes sense when under budgetary pressure, agencies will soon discover that it's not a sustainable approach.

The Challenges with In-House DevOps Implementations

To implement a DevOps strategy correctly within any agency or public entity requires a steep learning curve. By assigning DevOps related tasks to traditional software engineers, the project will often run into unexpected snags and encounter issues that an experienced DevOps consultancy would have easily avoided.

The primary benefit that comes from experience is knowing which tools best solve the problems faced during the DevOps lifecycle. If software engineers are manually provisioning infrastructure and updating configurations, the team will soon discover that variations and errors will creep into the system. The additional stress that comes from the constant context switching will also lead to an unhappy workforce, and the entity can expect an increase in staff turnover.

The Cost of Getting DevOps Right from the Start

DevOps provides massive ROI to organizations simply as it continuously eliminates repetitive tasks from the Application Lifecycle Model (ALM). When the number of hours spent on manually doing things starts decreasing exponentially, is when you know that your DevOps strategy is succeeding. This is exactly how organizations should approach the funding problem when justifying the cost of the additional resources.

A Puppet Report from The State of DevOps in 2016 found that high-performing organizations deployed 200 times more than their peers, and the average cost per downtime was $2.75M less than lower-performing entities (companies that deployed only seven times per year). With these kinds of ROI, finding money in the budget for an expert DevOps resource should be a priority.

The *Zero to DevOps Hero* Hiring Approach

The best approach for entities with zero expertise is to bring in a DevOps consultancy with years of experience. The consultant will assess the systems, processes, tools, and resources to build a roadmap for going forward. Once the roadmap is developed, the consultancy can augment the staff in order to build the solutions. As the embedded engineers work with the employees daily, in-house resources will gain both exposure and have access to expert advice on every element of the new tools and processes deployed.

For organizations that opt instead to hire a dedicated resource, the initial costs will be much greater. An experienced DevOps resource's salary can exceed $100,000 per year, with recruitment fees not included. For public entities with static budgets, it could easily prevent any new initiative from gaining ground. Estimations from the Puppet Report show that the timeframe with a dedicated in-house resource to provide adequate ROI to the organization is nine months.

Conclusion

Public entities face many of the same problems as the private sector, with additional overheads coming from government policies while working with additional budgetary constraints. Under these conditions, agencies may think that moving to an agile development model is not achievable. As previously described, that is clearly not the case. With the right partnerships and expert advice, a DevOps lifecycle implementation can be successful and improve the organization's services.

More and more government agencies are embracing a DevOps strategy to update technology and infrastructure. A continuous development methodology delivers code that is reliable and releases new features in hours instead of weeks. The digital transformation of the technology stack improves collaboration and teamwork and increases staff engagement and fosters a culture of participation throughout all the Business Units (BUs) and departments.

With the right tools and automated processes, every agency can deliver on the mandates expected by the public and oversight committees with greater efficiency and consistency.

ReleaseTEAM's Service and Solution Offerings

ReleaseTEAM is an agile, veteran-owned DevOps consultancy that has served in the trenches of the software revolution for over twenty years. Our consultants are well versed in the challenges facing public sector organizations with additional control and oversight requirements. As experts in the DevOps workflows and tools, by collaborating with ReleaseTEAM to modernize your agency, you can reduce the total cost, ensure the implementation's success, and improve the agency's ability to achieve its goals while keeping the entire system secure.

Software Sales and Support Services

ReleaseTEAM provides a comprehensive suite of solutions that will fit within any organization's operational frameworks. With complete pipeline solutions, project management tools, extensive automation, and secured code repositories, any agency can confidently move towards a modern digital technology landscape.

Staff Training and Mentoring

Every implementation of a new technology stack requires training and post-sales support. ReleaseTEAM provides staff mentoring, group training, as well as team augmentation with embedded DevOps engineers. When it comes to transferring the necessary skills for a new suite of products, ReleaseTEAM will assist your staff in taking on the daily challenges with confidence.

GSA and Development Outsourcing

When teams are under pressure, and there is a new initiative strategically important to the agency, ReleaseTEAM can assist with managing the entire project from concept to delivery. ReleaseTEAM's group of experts will deliver the software required within the parameters defined at the project's outset.

With GSA, the entire development ecosystem runs under ReleaseTEAM's purview and responsibility. ReleaseTEAM's resources strategically for timeous delivery will enable in-house teams to focus on mission-critical tasks and reduce undue pressure on the agency's staff.

About ReleaseTEAM

ReleaseTEAM is a veteran-owned, full-service DevOps consulting firm, servicing both the public and private sectors. We focus on helping our clients to adopt and mature their software development lifecycle methodology, with solutions tailored to their objectives and culture. We offer the best solutions in the industry, delivered with over 20 years of expertise.

DevOps consulting, support, staff augmentation, mentoring, and DevOps software solutions. We work with you to understand your unique needs, audit your current development process, and recommend adoption paths with the highest return. Our consultants then evaluate the tools and own the implementation.

ReleaseTEAM, Inc
3879 E. 120th Avenue, #154
Thornton, CO 80233

866-887-0489
info@releaseteam.com

More Free Resources from ReleaseTEAM

©2021 ReleaseTEAM