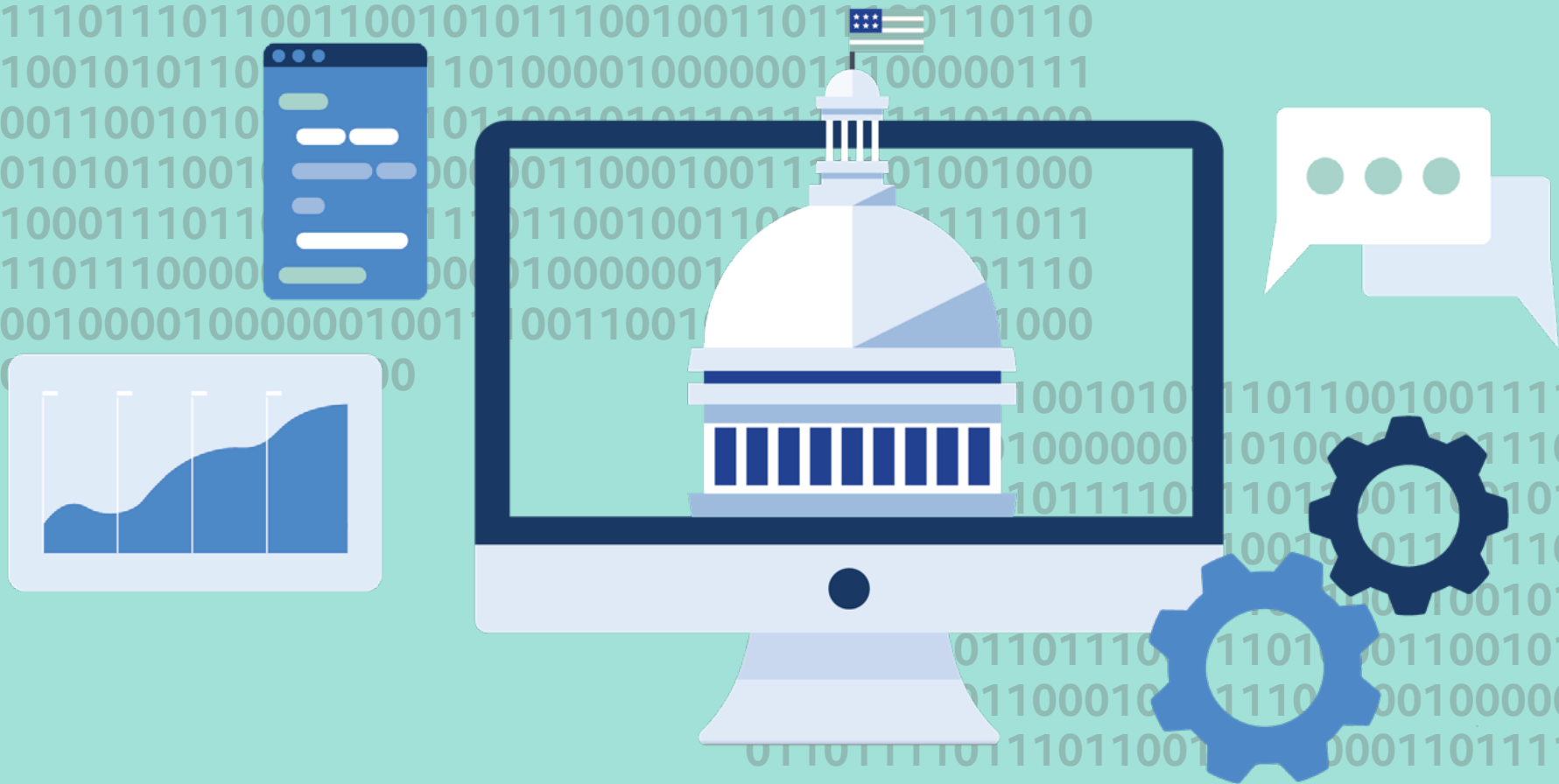


DevOps in Government



GOVLOOP
E-BOOK
2019

 NetApp®

 govloop

 TechData

Executive Summary

In any organization that creates its own technology applications or software, there are two broad teams that complete that task: developers and operations staff. Traditionally, these teams follow a linear development process. Developers create code for the service based on broad parameters provided by the organization's leadership. Once the developers build the product, they send it to the operations team to deploy and manage. This handoff is often referred to as "tossing it over the wall" because operations and development staff pass the product off without truly understanding how each team is using the application. It's like there is a wall between them that prevents transparency or meaningful communication. That wall creates all sorts of problems. Because developers don't have a clear view of how operations will use the service, the product they create often doesn't stand the test of real-world deployment. The new service can disrupt existing workflows, overload or damage current IT systems, compromise security or even fail regulatory requirements because developers didn't have enough information during development.

DevOps is a project management methodology that solves this problematic product handoff between developers and operations staff. Instead of operating in silos, both teams collaborate on a project from start to finish.

The phrase DevOps is simply a condensed version of the terms "development" and "operations," but what it actually means is not as simple. That's because it's an IT services management approach, and is related more to a cultural shift and organizational environment than technology. So instead of being a specific toolset, analysis or process, it's a more holistic approach and cultural mindset change.

For the purposes of this e-book, we are defining DevOps as an IT services management approach that stresses communication, collaboration, continuous feedback, experimentation and integration between software developers and IT professionals.

In the following pages, we offer practical tips to improve and achieve DevOps from government experts who are leading successful efforts at their organizations. We'll focus on four critical categories: culture, automation, measurement and sharing (CAMS), and how DevOps tools and technologies can aid in these areas.

These tools can make development and operations workflows more streamlined and collaborative, automating previously time-consuming, manual or static tasks involved in integration, development, testing, deployment or monitoring.

Additionally, we'll take a look at why DevOps matters to all government employees today, and how it affects end users and citizens; methods you can use to achieve DevOps; benefits of DevOps; and its history.

DevOps at a Glance

What impact does DevOps have on government? How does it help you achieve your agency's mission, and where did it evolve from? These stats will help set the context for why DevOps culture, technology and outcomes are more important than ever for governments at all levels.

75%

of CIOs identified DevOps as a top priority.

2,604x

How much faster advanced DevOps teams recover from IT incidents.

7x lower

failure rate on IT changes made by advanced DevOps teams compared to lower-performing teams.

2x

IT teams practicing DevOps processes are two times as likely to exceed their goals in profit, market share, and productivity.

80%

of government efforts to transform performance don't fully meet their objectives.

17%

of technology professionals said in 2018 that their teams have fully embraced DevOps.

80%

of federal IT projects were self-described as "agile" or "iterative" in 2017.

"Within 3 months, IT improved its time to resolution 66%, reduced call waiting time 50%, and increased customer satisfaction by 140%."

- Carol Johnson, IT director for Telegraph Media on DevOps strategies in 2016.

Source: Atlassian and HDI Trends You Need to Know: DevOps and IT Support

The 'Why' of DevOps for Government Today



What is DevOps?

As government transitions away from linear application development processes and into the new era of collaborative processes, application development is becoming quicker, cheaper and more efficient. Instead of operating in their traditional silos, developers and operations staff have begun to work together on projects from start to finish.

That's where DevOps comes in.

DevOps is an evolving philosophy and framework that encourages faster, better application development and faster release of new or revised software features or products to customers. It evolved from original agile practices in project management that were then applied to technical project management and IT projects. The practice of DevOps encourages smoother, continuous communication, collaboration, integration, visibility and transparency between application development teams (Dev) and their IT operations team (Ops) counterparts.

Benefits of DevOps

Collaboration — the very core of DevOps — is always a good thing. Bringing more minds to the table engenders more diverse ideas and helps focus teams on a common mission. Plus, it increases transparency across an organization because different departments and teams are working more closely together. But in addition to the basic benefits of collaboration, there are a ton of other reasons to use the DevOps process. It can help your agency achieve:

Quicker development: For many DevOpsers, this is the most important benefit of the process. With DevOps, the end goal is to develop and shape code more quickly and efficiently, and with more iterations. Because developers and operations teams don't have to waste time throwing code back and forth over a metaphorical wall, they can release iterations of their product sooner. And when they need to make changes or improvements to those applications, they can do so faster because both teams have a shared understanding of the project's functions and goals.

Greater quality assurance: But even as workers deliver more services faster, they don't have to sacrifice quality with DevOps. Actually, the process minimizes errors through two distinct attributes. First, DevOps allows both teams to see the full scope of a project. That means they have a better understanding of how any single change — say, a

change in a line of code or an update to a hosting platform — will affect the rest of the application. In traditional development processes, one change could derail an entire project. That's far less likely because of the collaborative workflow of DevOps. Second, automation replaces many human processes within DevOps. DevOps reduces human error by automating many core development and deployment functions.

Better end products: With DevOps, “we see higher-quality solutions, and at the same time we see flexibility,” said Texas.gov Director of Technology Peter Eichorn. When they incorporate DevOps practices, both developers and operations workers gain a better understanding of the product they're working on. Developers get to know the real-world conditions in which their code will operate. Operations teams learn what makes their services tick and why they were built that way. Together, both teams can make better decisions about how to build and deploy services, and that results in a better product overall. And because there are minimal barriers between the development and operations environments, employees can make changes with minimal stress to ensure that applications keep pace with evolving needs and scenarios.

Happier end users: Ultimately, DevOps helps government agencies better serve citizens in the digital age. As users increasingly expect government resources to be provided with the same mobility, agility and utility as private sector services, agencies seek ways to quickly meet those demands. When they turn to DevOps, they're able to deliver those services and meet citizen expectations.



CAMS and DevOps

Some people group DevOps goals into four categories: culture, automation, measurement and sharing. DevOps tools can aid in these areas. These tools can make development and operations workflows more streamlined and collaborative, automating previously time-consuming, manual or static tasks involved in integration, development, testing, deployment or monitoring.

Let's take a closer look.



Culture. In order for DevOps to succeed, technical, regulatory and cultural government silos must be significantly reduced. A DevOps culture is collaborative and employees across departments are in constant communication and embrace rapid change.

Automation. The replacement of manual processes to accelerate digital product delivery is another core tenet of DevOps, but quality and security should not be sacrificed. Automation improves efficiency and reduces the potential for human error.

Measuring. By measuring the outcomes of your projects and how effectively you got there, you can prove the value of the process to others in your agency and make process changes as needed.

Sharing. Information-sharing improves efficiencies by reducing duplicative efforts. Employees can focus on mission-critical issues and achieve common understanding of the processes and goals that have been set.

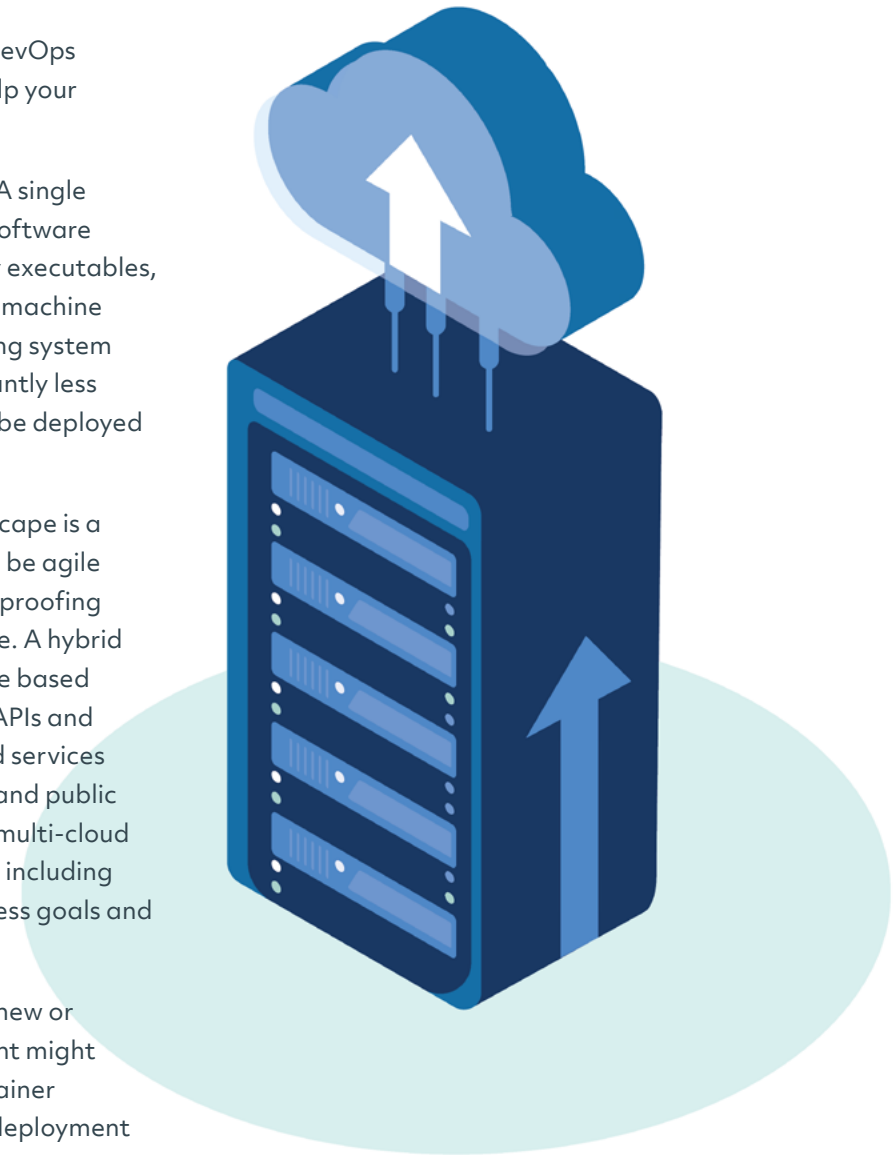
Tools and Technology to Help Support DevOps

As we've stated, there is no one single tool or technology that is a full DevOps solution. But used in tandem, a variety of tools and approaches can help your agency actually implement DevOps. Here are a few key ones.

Containers: Containers are a form of operating system virtualization. A single container might be used to run anything from a small microservice or software process to a larger application. Inside a container are all the necessary executables, binary code, libraries and configuration files. Compared with server or machine virtualization approaches, however, containers do not contain operating system images. This makes them more lightweight and portable, with significantly less overhead. In larger application deployments, multiple containers may be deployed as one or more container clusters.

Hybrid multi-cloud experience: Understanding the new DevOps landscape is a good first step, but it's important to act quickly. Organizations need to be agile enough to say no to previous investments, and instead invest in future-proofing their DevOps culture by moving toward a hybrid multi-cloud experience. A hybrid multi-cloud experience allows an organization to provide an experience based on frictionless consumption, self-service, automation, programmable APIs and infrastructure independence. And it allows them to deploy hybrid cloud services between traditional and new applications, and between data centers and public clouds quickly and efficiently. It's not an either-or scenario. The hybrid multi-cloud experience allows organizations to leverage all available technologies, including new innovations and traditional legacy applications, to advance business goals and drive success.

Continuous deployment (CD): This practice automates the release of new or changed code into production. An agency doing continuous deployment might release code or feature changes several times per day. The use of container technologies, such as Docker and Kubernetes, can enable continuous deployment by helping to maintain consistency of the code across different deployment platforms and environments.



Supporting the Modern Warfighter With DevOps Initiatives

Today's DoD software development and deployment is working to become more responsive to warfighter needs. In the Air Force, they're using modern software development techniques and processes like DevOps to keep up. **Nicolas M. Chaillan**, Chief Software Officer, United States Air Force, spoke with GovLoop about the DoD Enterprise DevSecOps Initiative, which is focused on bringing automated software tools, services and standards to DoD programs so that warfighters can create, deploy and operate software applications in a secure, flexible and interoperable manner.



GOVLOOP: As a DevOps professional in government, what are the issues that are top of mind for you today?

CHAILLAN: There are two sides for us today that are top of mind. There is the talent side, and then the technology side. A lot of people say the culture is the hardest – I agree, but also, part of me really thinks the technology is not that easy. In terms of talent, being able to pay the talent that we need to compete with the biggest companies on earth for DevSecOps talent is hard.

On the technology side, for us at the DoD, it is unique. We refer to DevOps as DevSecOps because of the security component for us. So at the DoD we have applied DevSecOps to 29 weapon systems in our DoD Enterprise DevSecOps Initiative, which is focused on bringing automated software tools, services and standards to DoD programs. The goal is to expand to 172 in the next six months. Using this approach is complex – it's a little bit harder when it's a weapon system, but we still do DevSecOps with the weapon systems. So it's pretty unique, and exciting.

You know technology-wise, we built our DevSecOps initiative to avoid reinventing the wheel. For example, we use containers, to be able to accredit software and tools once across DoD, and then be able to use it across the enterprise. From there, people could take

the container, launch their project and use it in different ways.

And the use of open source containers in DevOps is critical for us. Importantly, the use of open source containers will reduce the need for us to be locked into one vendor's technology.

Could you speak a bit more on how the technology that you work with enables DevOps?

So one of the biggest wins we've built was to have this centralized repository of containers where we take products and we secure them. It's called hardening. So we harden the container. We take it from whatever vendor or open source community and make it a bit more secure, and then we store that into the central repository of containers, that all the teams and the 29 initiatives that I told you about can come and use.

So it becomes Lego blocks that they can use to set up their factory, to set up the tools or also to set up their own applications. It's all containerized – we don't run different machines. It's also all open source. So we're also trying to show that there's not a secret sauce to DevSecOps. We here at the DoD might be a little bit more secure than the usual team, but it's just a matter of dedication and best practices.

So why is DevOps especially important now, and how do you see the landscape changing in the future - especially at DoD?

I think we just cannot afford not to use it. Things move too fast, and anything that slows you down to get your job done, to build your mission application, to build what you're here to do, and get it done fast, you have to move past it. DevOps helps us to learn fast, fail fast, and it's OK to fail, as long as it's fast and not twice for the same reason again. And when you're trying to innovate, to do record prototyping and build the next cherished weapons, or business systems, anything that's related to the duty of mission, you need AI, you need machine learning, you need modern

compute and cloud, and if you don't have DevSecOps, you won't be able to do it.

And then with the landscape, I think people are often mistaken when they talk about government employees being afraid to do something like this and bring in change. It's not true at all. I think everyone is pretty excited to do better and faster and, and they all jump on the occasion. It's just a cultural change, which I admit, can be hard. You know when you're used not to doing something, and you don't know better, and you don't know there's another way, you just do what you know. But once you show government employees the light, and how it makes improvements, I think people are pretty excited and more than happy to jump on it.





DEPLOY THE RIGHT FOUNDATION FOR MODERN APPLICATION DEVELOPMENT.

Across all clouds — public and private.

Explore our DevOps solutions →

Shifting Both Technology and Culture Towards DevOps

An interview with Ingo Fuchs, Chief Technologist, Cloud and DevOps, NetApp

The sands are shifting in the technology that supports government today. In order to better serve citizens, keep critical data safe, and deploy larger scale web applications, government IT teams have had to shift not just their technical approaches and tools, but also their culture. To keep up with an evolving world and citizenry, agencies need to automate and quickly spin up or tear down production-quality development and test environments at will.

That is where DevOps comes in. DevOps opens the door for continuous innovation, rapid software deployment, and frequent updates to software-based features and products.

Part of that success depends on how well DevOps is fueled by the organization's underlying data center infrastructure and processes. To better understand how an innovation-minded DevOps focus translates into practical moves and investments that agencies should make, GovLoop spoke with Ingo Fuchs, Chief Technologist, Cloud and DevOps, at NetApp, a leader in hybrid cloud data services and data management.

Fuchs explained that the long-standing cultural divide between development and operations has traditionally inhibited collaboration and led to slow development cycles.

"Government digital services teams should consider using open source, cloud-based, and commodity solutions across the technology stack, because of their widespread adoption and support by successful consumer and enterprise technology companies in the private sector," Fuchs said.

The culture of DevOps aligns itself with self-service provisioning and automated processes that are usually associated with the cloud, he continued.

By creating data services that help build trust between developers and operations, NetApp aims to provide the tools that, together with DevOps techniques and methods, enable companies to transform this paradigm.

NetApp is no stranger to DevOps. While most federal agencies have historically known NetApp as an enterprise storage provider, NetApp has also spent the past several years transforming into a data authority.

"NetApp's data management expertise can be leveraged easily with DevOps tools—from containers to automation to hybrid and multi-cloud," Fuchs said. "Our solutions and integrations go across the entire spectrum of DevOps and hybrid cloud."

"We have been committed to DevOps, containers, automation, and more, for a very long time. This means we are uniquely positioned to help government adopt hybrid and multi-cloud models to accelerate DevOps in their own agencies."

-Ingo Fuchs, Chief Technologist, Cloud and DevOps, NetApp

Beyond just technology, NetApp's own IT organization transformed its culture and its implementation from a traditional, multi-datacenter environment to a DevOps-friendly environment, and their IT built an internal DevOps platform, called CloudOne, that provides the cloud services, automation, and CI/CD release models used by their application development teams to build cloud native applications.

Just like federal civilian agencies, Fuchs explained, over the next few years, NetApp IT will migrate a large number of its applications to SaaS, replacing legacy apps per an application rationalization model. The remaining applications will move to the CloudOne platform which provides multi-cloud integration and DevOps services to rebuild and run them as modern cloud native applications, he noted.

Takeaway: In order to better serve citizens, keep critical data safe, and deploy larger scale web applications, government IT teams have had to shift not just their technical approaches and tools, but also their culture to enable DevOps.

A man and a woman are working together at a computer in a modern office setting. The man is pointing at the screen while the woman looks on. They are both focused on their work.

Cloud, DevOps, and State and Local Government

How effective can DevOps be when it comes to state and local? GovLoop asked **Pete Eichorn**, Director of Enterprise Architecture for NIC state, local and federal solutions,. He says cloud, contracts, and balance are key.

GOVLOOP: What's happening in terms of DevOps at the state and local level that you see out there?

EICHORN: The macro trend that's been coming and is in full force — especially with state, and I think to some degree local — is cloud. Cloud, for me, is the vehicle, if you will, that makes so much of DevOps work. And with the trend of using DevOps, we can have increased adoption with cloud and all of its benefits. Better capabilities, including advanced innovations like artificial intelligence and machine learning, are things that we're truly now adding into products and services.

I think the other trend, if you go a little bit deeper, is that the tools and the technology specific to DevOps are just so much more advanced, that they only continue to get easier for techies to use. And along with that, these tools can provide better controls and better governance than anybody can really do manually. This means the tools and the automations that go with building software, pipelines, technical operations, security and more; all of these are becoming more automated and benefiting DevOps approaches.

What challenges might you be seeing when it comes to DevOps adoption?

In most cases it is really the skills, and teaching the right skills, to the right people. Agencies really have to find ways to teach people and grow people into these skills that are required for DevOps. So that's probably one of the bigger challenges, and the good news is that cloud providers often provide training and ways to self-service, and even certifications that you can get. That can help make the skills needed to work within DevOps more accessible.

What role are DevOps and IT modernization playing together?

For me, modernization comes in a couple different dimensions. One is the topic of consumerization of technology, in which citizens expect their government to provide many capabilities that are easy to use. I think that DevOps plays an important role, because citizens expect updated features to come out regularly. DevOps is the methods and techniques that allow you to deploy things more easily on a regular cadence, and it makes the whole experience better as the software improves over time.

I'd say the other angle is about dealing with legacy systems. There, DevOps is important because it allows you to do things at an

incremental pace. So rather than facing a large multiyear, all-or-nothing modernization effort, you can really modernize existing systems with improvements and you can add new capabilities alongside existing ones. DevOps is really good for that because of the incremental nature of its approach.

DevOps is often referred to as a way of life rather than any sort of institutionalized policy. But do you see anything moving towards the maturing of policies and standards around DevOps?

I think it's moving forward carefully in that direction. One example I would compare it to is the evolution of cloud. For the procurement of Cloud services, the language in contracts has changed as the technology changes, including a more flexible mindset that DevOps requires. So I'm actually noticing that changing, and it's less prescriptive on some of the controls that are typically in contracts. This can allow agencies to modernize in a faster way. So, as I see it, that's the way policies (as they're reflected in contracts) seem to be moving forward in the right direction.

As you continue to see DevOps evolve in government, what is coming to mind for you?

I think it's important to keep in mind that there's always a balance, especially meeting government's needs. We are balancing all the traditional aspects of what works with project management and governance with the latest technology capabilities. Contractual requirements and governance are also in play when you're looking to adopt techniques such as DevOps.



Conclusion & Next Steps

Before agencies dive into the next stages of their DevOps journey, they must understand a variety of technologies, practices and approaches to changing the culture that support the way forward.

Creating a Culture for DevOps

Implementing DevOps in your organization isn't as simple as buying a tool and making sure that everyone uses it. DevOps is more than a person or even a team of resources; it's a culture of constant evolution. In essence, it's not a destination you arrive at, it's a journey of improvement. So how can you create a culture of DevOps? Here are a few ideas.

Improve communication: Communication is key. Because DevOps is iterative, and there isn't a true endpoint, you need to share metrics and empower your teams, motivating them to move forward. Be open to new ideas and encourage outside-the-box thinking across all levels of your organization to encourage the innovation that will drive your business forward.

Alleviate fears of change: Remember that team members may fear this change. Shifting roles and new responsibilities can create discomfort and uncertainty. Sharing the vision of what their jobs will look like in a DevOps environment and providing the training to set them up for success (both in your organization and in their careers) will help alleviate some of the growing pains that can hamper your progress.

Give your team the right infrastructure:

Without the right architecture in place to build, manage and distribute your data to the people and places it needs to reach, you won't meet your goals. You need to evaluate your current infrastructure and identify your gaps. Move toward an environment that can support the automation, integration and monitoring you'll need to be more agile and efficient wherever your business requires it, on-premises or in the cloud.

Glossary

Here is a glossary of DevOps practices that reflect the idea of continuous improvement and automation. Many practices focus on one or more development cycle phases.

Continuous development. This practice spans the planning and coding phases of the DevOps lifecycle. Version-control mechanisms might be involved.

Continuous testing. This practice incorporates automated, prescheduled, continued code tests as application code is being written or updated. Such tests can speed the delivery of code to production.

Continuous integration (CI). This practice brings configuration management (CM) tools together with other test and development tools to track how much of the code being developed is ready for production. It involves rapid feedback between testing and development to quickly identify and resolve code issues.

Continuous delivery. This practice automates the delivery of code changes, after testing, to a preproduction or staging environment. A staff member might then decide to promote such code changes into production.

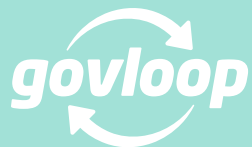
Continuous monitoring. This practice involves ongoing monitoring of both the code in operation and the underlying infrastructure that supports it. A feedback loop that reports on bugs or issues then makes its way back to development.

Continuous deployment. Similar to continuous delivery, this practice automates the release of new or changed code into production. A company doing continuous deployment might release code or feature changes several times per day. The use of container technologies, such as Docker and Kubernetes, can enable continuous deployment by helping to maintain consistency of the code across different deployment platforms and environments.

Infrastructure as code. This practice can be used during various DevOps phases to automate the provisioning of infrastructure required for a software release. Developers add infrastructure “code” from within their existing development tools. For example, developers might create a storage volume on demand from Docker, Kubernetes or OpenShift. This practice also allows operations teams to monitor environment configurations, track changes and simplify the rollback of configurations.



Thank you to NetApp and Tech Data for their support of this valuable resource for public sector professionals.



About GovLoop

GovLoop's mission is to inspire public sector professionals by serving as the knowledge network for government. GovLoop connects more than 300,000 members, fostering cross-government collaboration, solving common problems and advancing government careers. GovLoop is headquartered in Washington, D.C., with a team of dedicated professionals who share a commitment to the public sector.

For more information about this report, please reach out to info@govloop.com.

govloop.com | [@govloop](https://twitter.com/govloop)