ACCELERATING YOUR DEVOPS JOURNEY IN THE PUBLIC SECTOR

redhat

In the public sector, programs can run slowly and inefficiently. The government's information technology systems are growing more outmoded while the public's expectations about technology, speed and delivery are rising. As the private sector pushes IT boundaries and experiments with improved services, the public sector faces challenges of regulations and constrained financial resources. Additionally, unlike the cutting edge firms in the private sector, the public sector is saddled with legacy IT that they have to modernize. True, federal agencies don't necessarily need to be Google or Amazon – but many underestimate the significance of IT in executing on their missions. Today, every company today is a technology company regardless of what line of business they are in.

Bottom line: Government needs more efficient IT management systems.

Fortunately, recent innovations and developments in technology and IT management provide opportunities to help the government deliver its technological products more effectively.

In particular, the public sector has begun to look toward a project management and software development approach and mindset known as DevOps to cope with some of these challenges.

This Industry Perspective aims to provide you with the basic tools necessary to start implementing DevOps for your IT project management. Although it's perhaps intimidating and seemingly hard to grasp at first glance, DevOps is a clear and common sense approach to technology.

We sat down with Evong Nham, Senior Solutions Architect at Red Hat, and Peter Chin, Application Development and Architecture division Chief at the US Courts, Department of Program Services, to hear about DevOps and its implications and utility for government organizations. We'll guide you through the basics, and by the end of this Industry Perspective, you'll have all the essential information to start exploring using DevOps in your agency.

DEVOPS: A QUICKER, MORE EFFICIENT APPROACH FOR IT

The phrase DevOps is a shortened, condensed version of the terms development and operations, but it's often thought of as difficult to define. That's because it's more of a cultural shift and organizational environment than a specific type of analysis, process or toolset, and anything related to culture is notoriously hard to define. DevOps defies specified boundaries and limits, and seeks to be a more holistic approach to technology.

So where to begin? Every DevOps expert and user has his or her own definition, but most can agree on some basic facts of what it is — and what it isn't. Very broadly, **DevOps is a software development method that stresses communication, collaboration, continuous feedback, experimentation and integration between software developers and IT professionals.**

Ultimately, the approach is all about delivering software more quickly and effectively to the user. Its aim is to help an organization swiftly produce software products and services to save time and money by removing the barriers between the development and delivery of a software product.

In DevOps, product development is constantly happening, and there is constant feedback and improvement, no matter where in the process a particular product stands. Technology products begin to be used while they're being developed. In some cases, that means bringing them to market very early in the process, but sometimes that just means using them internally within the organization.

Once the products are being used, DevOps strategy encourages feedback from users so the product can be reevaluated. Changes are then implemented and the updated product is delivered. Basically, the delivery times are shortened a great deal and re-peated multiple times. Software becomes a work in progress.

Throughout the process, automation is deferred to whenever possible. It allows for increased efficiency by letting computers take over the mindless work whenever possible.

But more than anything, DevOps is a culture. It's a radical new shift from the older approach to management that values longstanding traditions and established methods. It seeks to change attitudes so that failure is acceptable and innovation is celebrated.

"[DevOps] has given birth to a culture," Nham said. "It's become a lot more than common sense ideas or isolated tools. It's become a culture that spans all of the layers of the IT organization, like people, process and technology. It's impactful to all of those."

In DevOps, separate teams aren't so distinct. There is a general shift toward the attitudes that all people in the organization are working together to deliver an ultimately flawless technological product. Departments and teams shouldn't be completely distinct, but rather the organizational structure should foster cross-communication and interdependence.

The term DevOps refers to the bringing together of development engineers and professionals on the operations side. Communication and collaboration is a huge priority in DevOps and this is reflected in its name.

DevOps is often juxtaposed with Agile project management because the two are related, but they're not the same. Agile is a specific software development methodology, whereas DevOps has a much broader scope. Agile software development has a specified manifesto, checklists and a certifi-cation process attached to it, while DevOps is an entire IT strategy that affects a whole organization more fundamentally.

"DevOps is a broader initiative. It's about realizing efficiency across the entire organization," Nham said. "It's got the entire craft of IT. So, Agile may fall under DevOps, and it can certainly be one aspect to your DevOps strategy, but DevOps has way broader implications than just software development alone."

DevOps is also strongly connected with open source, given the proclivity both have toward constant improvement and feedback. "Open source and DevOps certainly share the same cultural roots," Nham said. "Open source and DevOps, just by nature, give you freedom of choice and flexibility to implement the right tool for your organization."

> **"[DevOps] has become a lot more than common sense ideas or isolated tools. It's become a culture that spans all of the layers of the IT organization, like people, process and technology. It's impactful to all of those."**
>
> *- Evong Nham, RedHat*

WHY DEVOPS?

Now that we know what DevOps is, we can delve a little deeper and ask: Why should government use DevOps? How can using this approach help government do its job better?

In traditional software development, known as the Waterfall approach, there are some inherent problems that DevOps hopes to solve. First, Waterfall isn't a constant learning- or improvement-centered approach. Lessons are gleaned at the end of development, rather than on a more constant basis. In general, it's a far slower process and delivery times are much longer than with DevOps.

Additionally, Waterfall is less cyclic than DevOps is. In Waterfall, the incorporation and implementation of constructive feedback doesn't happen as frequently or effectively. Rather, changes happen much more slowly and tend to be longer-term and more permanent, which can stifle improvement.

"In the Waterfall process, there's never any opportunity to iterate," Nham said. "We never build any feedback loops into the process. So we never have the opportunity to learn from our mistakes and get better. We set a course at the beginning and we never deviate from it, and we never challenge ourselves to find better ways of doing things."

In government, the name of the game is efficiency and delivery. The mission and functioning of the government affect the entire country, so it's important that things run smoothly and quickly. DevOps helps that happen.

Many agencies look at the DevOps "unicorns" – the Facebooks and Twitters of the world – and are overwhelmed by the amount of effort it would take to modernize their organizations to the point where they could operate that efficiently. But DevOps is not just for those unicorns. DevOps advocates an incremental, iterative approach to realizing those efficiencies. Have a long-term goal in mind, but start small – demonstrate quick wins, validate those with metrics, celebrate your wins and learn from your failures.

When efficiency is such a priority, DevOps becomes the common sense approach to IT development. DevOps streamlines IT development and enables it to constantly improve. Rather than giving time to each specific new change, the idea with DevOps is to fail fast, but to learn from those mistakes, allowing for speed and continuous delivery.

This increased efficiency means money and time are maximized. Additionally, DevOps allows the government to deliver cutting-edge platforms and services to the public faster than ever before.

CHALLENGES TO USING DEVOPS

If DevOps is so fantastic, why aren't government officials automatically and wholeheartedly embracing it? There are several challenges that people need to be aware of when dealing with DevOps.

FUNDING

First, funding is a major challenge that DevOps users must grapple with. Adequate funding in the government is always a point of contention, and many feel that budgets are shrinking. Especially compared to some of the major technology companies, the necessary resources simply aren't available, and this must be taken into account when implementing DevOps.

After all, technology is expensive and can be seen as a strain on the already tight resources. Professionals often view innovative IT management approaches as resource-draining, but in the long run, using DevOps helps organizations save money through improved efficiency. So, in terms of money, DevOps is actually the way to go.

TRADITIONAL IT SYSTEMS

The government must also work within the confines of legacy IT systems that need upgrading and an inability to make the optimal modernizations immediately, again because of the constrained resources. These old systems take capital to upgrade. While this is almost unanimously a good idea in terms of long-term investment, it can be very hard to come up with the upfront costs and convince leaders to make these investments an IT budgeting priority.

"Unlike the Silicon Valley firms, we don't have infinite revenue streams," Nham said. "It's not at our discretion to basically blow everything away we have and just implement new people, processes and technology."

CULTURE

Most of the challenges, however, simply come back to the notion of DevOps as a culture. Government tends to be risk averse because the implications of its work are so widely reaching. As a result, it can be hard to convince major players to take chances with a new approach — especially one that champions failure as a part of the process.

The problem with changing organization culture is that it takes time, patience and vast restructuring. It must be done structurally, with the leaders aligning steps and incremental changes. Also, managers must encourage their employees and communicate with them on how the change is going. Many people are afraid of change, especially if they have been in the business for a long time. Changing IT systems is actually the easy part. It takes a great deal more work to change attitudes and philosophies.

External help can ease the transition. Red Hat Consulting Services can help your organization deal with some of these adjustment processes. "With Red Hat Consulting, they've basically created a series of tailored offerings, workshops, etc., that can help your organization understand and assess the impacts and opportunities of implementing DevOps across your IT organization — the people, process and technology impacts of that," Nham said. "They're doing what they can to not only facilitate the technology aspect of it, but take that entire holistic approach." That helps with the culture shock.

CASE STUDY
IMPLEMENTING DEVOPS AT US COURTS

To learn a little bit more about how DevOps is being implemented on the ground in the public sector, we sat down with Peter Chin, Application Development and Architecture division Chief at the US Courts, Department of Program Services.

In the US Courts, Chin's responsibility is developing operational solutions and services to serve judges, clerks of courts, public defenders, and pretrial and probation officers.

"Where DevOps helps me," Chin said, "is in achieving my goal of supporting the US courts system efficiently through eliminating waste, identifying repeatable steps, and automating those steps."

Chin explained that the implementation of DevOps is not necessarily primarily an IT solution. He said rather it almost works backwards. If you can create a culture in a workplace that leads employees to want to try DevOps, that in and of itself is a success, because it means you have worked towards creating a culture where innovation and efficiency are prized.

"Previously as a government executive and also as a private industry consultant, I focused primarily on customer satisfaction and high morale," Chin said. "A couple of years ago the U.S. courts began adopting agile development best practices, and that was put in place primarily to try to get more valuable releases out the door. But along with that, what I wanted to do is really complete the cycle and receive and implement continuous feedback from our custom-er base. And part of that is implementing DevOps here. We realized that we needed to implement continuous integration and continuous delivery in our practices in order to achieve that. But the initial focus is on people and organization, and not just tools and techniques. We want to actually create a culture where people feel free to innovate."

An implementation of DevOps also naturally leads to more cross-department collaboration, Chin advised.

"What it really means is, we don't want to actually just make changes to production haphazardly," Chin explained. "We want to be able to make changes, but also manage the risk as we're rapidly changing the production environment. Part of building that culture is breaking down silos, and having people realize that there's some common things that people across departments all want to build and test as we're developing a lot a different things."

Chin stressed that even non-technical people can be made to understand the importance and benefits of DevOps, as its implementation can benefit all teams and customers.

"One of the first things I needed to do here when I came on board was to really un-derstand who my audience was, because that was the basis for how and why we needed to implement DevOps," Chin explained. "For the technical folks here, I used different points to sell this concept. One point was that having continuous software delivery through automation meant we'd have less complex problems to fix because there'll be smaller and incremental changes, and also we'll have faster resolution of problems. And then of course I had to sell this to our executive level that was only interested in the business benefits. And I explained that use of DevOps meant faster delivery of features, a more stable operating environment, and more time available to add value, rather than having to fix and maintain things all the time."

Chin ended by explaining that a final reason his agency is implementing DevOps is to break down silos through a reorganization.

"With that reorganization, we're streamlining our application portfolio," Chin said. "After we do that, we want to find ways to get feedback from the customer earlier on. And the way to do that is to have a continuous feedback mechanism through a DevOps implementation."

THE DOS AND DON'TS OF DEVOPS

Despite challenges, DevOps, if implemented strategically, can help speed and efficiency in IT development. So, how can it best be rolled out? Nham and Red Hat subscribe to some DevOps dos and don'ts that help.

☑ **DO move away from the monolithic in terms of architectures, projects and teams.** Not only is a monolithic approach fraught with delivery risk, it's also a lot harder to implement in DevOps practices. The establishment of strong communication skills and cross-functional teams is much harder when your teams consist of hundreds of people. Developing effective automated testing suites is much more difficult when the functionality to be tested is immense. Provisioning of environments to support traditional, monolithic workloads is cumbersome and doesn't provide the agility you need to keep up with market demand.

✖ **DON'T do something manual more than twice.** If you have a commonly recurring task, look to automation. Let the computers take care of business by automating low-value, repetitive tasks. Save workers time by allowing the technology to take over. Favor on-demand, self-services approaches wherever possible – eliminate the ticketing queue. Tools like IaaS, PaaS, and CI/CD platforms are your friends.

☑ **DO build meaningful automation by establishing standards first**. Define standard parts and processes before you attempt to create a supporting infrastructure of automation. Otherwise, your automation is useless because it doesn't handle the one-offs the customer requires or you end up managing many permutations of scripts.

✖ **DON'T reinvent the wheel.** Your organization should be focused on providing business value to your customers, not on being a software vendor. Don't get in the game of building a custom automation framework that you need to forever maintain and manage. Empower your teams to concentrate on delivering value by encouraging the use of industry tools and frameworks.

☑ **DO instill pride of ownership in your teams**. Having a process or team structure where there are points at which the deliverable is thrown over the wall encourages members to be disengaged and feel no accountability for the end product. Try a strategy in which a team has ownership of a product through to production and has clear visibility into the status at all points in the lifecycle. Make sure to include all team members in any celebration of wins to drive that pride of ownership.

✖ **DON'T promote a culture of secrecy, judgment and fear of retribution.** Try to create a culture in which an individual is not scared to admit a mistake publicly. Institutionalize blameless post-mortems so teams feel comfortable discussing what lead up to a production incident and collaboratively learn how to avoid it in the future. Establish a relationship of trust between managers and developers or IT operators — things like judging developers by numbers of lines of code, attitudes toward failure and not being transparent about goals or motivations are counter-productive.

☑ **DO encourage innovation and experimentation.** Find ways to reduce barriers to entry for teams to enjoy creativity. Allow people to experiment and fail, and don't shame or blame them if they do.

✖ **DON'T create experts.** Avoid people bottlenecks by creating a single point of failure on your team. Furthermore, instilling in team members at least a high-level concept of each of the roles, their function, and the tools and processes used promotes transparency and understanding.

☑ **DO embrace feedback, metrics, and monitoring — it should be one of the first things you address.** To make impactful changes, you must establish a consistent feedback loop to identify pain points and obstacles. A good operational awareness needs to be easily accessible and shared by all team members so that they can feel ownership and make informed decisions.

✖ **Don't get overwhelmed – it will probably get worse before it gets better.** As you identify and address bottlenecks, you may uncover bigger, more difficult constraints. For example, you may be bottlenecked in access to test environments and you have testing teams sitting idle waiting for access. Once you fix this problem, you may realize that now manual testing is the constraint – taking 2 weeks to validate a bug fix may lead to developers who are now sitting idle. A knot is unraveled one kink at a time – stay focused on finding and eliminating kinks.

> **"The point that DevOps is trying to achieve is more efficiency — better quality, consistency and repeatability over the whole craft of IT. And I think that's something that every organization, big or small, public-sector or private-sector, can benefit from."**
>
> *- Evong Nham, RedHat*

RED HAT + DEVOPS

Rapid innovation, changing business landscapes and new IT demands force business-es to make changes quickly. The DevOps approach is a way to increase business agility through collaboration, communication and integration across different teams in the IT organization. And Red Hat offers consulting services that can get you there.

By aligning development and operations teams with solutions from Red Hat Consulting and OpenShift Enterprise by Red Hat, you can reduce product cycle time, write applications faster and drive automation that strengthens the link between IT and the needs of your organization.

OpenShift Enterprise by Red Hat is an on-premise, private Platform-as-a-Service (PaaS) solution offering that allows you to deliver apps faster and meet your enterprise's growing application demands. With Red Hat's award-winning private PaaS OpenShift Enterprise, you can improve developer productivity, increase operational efficiency and expand your hardware utilization.

redhat®

ABOUT RED HAT

Red Hat® is the world's leading provider of open source solutions, using a community-powered approach to provide reliable and high-performing cloud, virtualization, storage, Linux®, and middleware technologies. Today, Red Hat is at the forefront of open source software development for enterprise IT, with a broad portfolio of products and services for commercial markets. That vision for developing better software is a reality, as CIOs and IT departments around the world rely on Red Hat to deliver solutions that meet their business needs. Solutions that provide technology leadership, performance, security, and unmatched value to more than 90% of Fortune 500 companies.

ABOUT GOVLOOP

GovLoop's mission is to "connect government to improve government." We aim to inspire public sector professionals by serving as the knowledge network for government. GovLoop connects more than 200,000 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 connect and improve government.

For more information about this report, please email us at: info@govloop.com

1101 15th St NW, Suite 900
Washington, DC 20005

Phone: (202) 407-7421 | Fax: (202) 407-7501

www.govloop.com
@GovLoop

govloop

1101 15th St NW, Suite 900
Washington, DC 20005

Phone: (202) 407-7421 | Fax: (202) 407-7501

www.govloop.com
@GovLoop