YOUR 2020 GUIDE TO A

Smarter DevOps

Discover more resources
at nvisia.com.

Contents

Speed, Stability, and a Culture of Innovation

The Power of DevOps

DevOps is a powerful tool that promises speed, stability, and a drive for continuous improvement and innovation. It's workflow, technology, and culture wrapped into one.

Just how fast? Try **106% faster**, from lead time to deploy, with more reliable results, too (*The State of DevOps Report, 2019*).

You've likely heard of DevOps before and may already be implementing it within your organization. Whether you're already familiar with it or just dipping your toe in the water, this guide is here to help walk you through a broader scope, how to make the most of DevOps, and why it's important to reach your full DevOps potential.

After a high-level overview that delves into all DevOps can do when approached through a 360 degree lens, this guide walks you through actionable ways that you can improve DevOps practices within your organization—culturally, and by employing a backlog, a capability matrix, the Craw-Walk-Run Model, and gauging key metrics. Finally, you'll come to the Action List that outlines your next steps towards a smarter DevOps.

It's common sense for anyone in software.

Today, every company is a software company. In the digital era, delivering effective, up-to-date products to customers is critical for businesses. From web clients for consumers to applications that enable company employees to do their jobs efficiently, software products drive your company's success.

In a world that evolves at breakneck speeds, you cannot afford to wait for long software release cycles.

Getting new, relevant features in front of users quickly is key in gaining an edge over competitors. DevOps is all about doing exactly that: getting software products to market quickly. Not only does a more effective practice bring speed to your software releases, but it contributes to stability and confidence in what is released, as well.

At its best, DevOps encompasses the whole of your IT department. It relies on efficient product management to discover software product features, and teams that get things in front of your users as quickly as possible. Time is spent innovating, experimenting, testing, and problem-solving in a continuous flow. Not a moment goes to waste.

Consider COVID-19

The business landscape changed almost overnight. Educational institutions and offices alike were forced to migrate to an entirely digital environment. In acknowledgment of this paradigm shift, products like ZOOM and Microsoft Teams excelled, adding new features to support a remote workforce, online education, & other pandemic-related communication.

However, many institutions that struggled to adapt to the rapid change, such as schools that suffered record-high absence rates while attempting distance learning, found themselves unable to thrive in lieu of the crisis *(Virus Forced Schools Online, but Many Students Didn't Follow)*. DevOps enables quick action that can make or break an organization in such a fast-paced world.

The DevOps Way

DevOps isn't just a workflow; it's a mindset.

DevOps is not just about upgrading to better tools or a streamlined workflow; this is about a culture and attitude that spans all areas of software creation and implementation within your organization.

While it can help with specific products a team is working on, DevOps also has the power to streamline your entire IT department. By automating repetitive tasks, like tickets and changes, developers are able to spend their time on what they're passionate about: writing great code for challenging projects—ones that contribute to business value. The results are resilient, manageable, and observable, creating high-impact changes and frequent releases.

So, What Does the Best of DevOps in Action Look Like?

Per the 2019 State of DevOps Report, "Cloud continues to be a differentiator for elite performers and drives high performance." There is a strong correlation between high performance organizations and the use of cloud native technology. The best tech of today and the standard of tomorrow lies in the cloud. To make the most of DevOps and compete with top industry performers, it's a foundational necessity.

The Tenants of DevOps

Flow

From beginning to end, flow is increased by removing constraints and reducing WIP.

Feedback

Create tight feedback loops with automation and testing to catch errors quickly and continuously correct them.

Continuous Improvement

Adopt an attitude of experimentation and practice where it's encouraged to take risks, learn from mistakes, and hone skills for mastery.

What are the benefits of cloud native technology?

- Cloud native technology empowers organizations to build and run scalable applications in modern, dynamic environments. It employs containers, service meshes, microservices, immutable infrastructure, and declarative APIs.
- It enables loosely coupled systems that are resilient, manageable, and observable, allowing high-impact changes frequently
- It fosters and sustains an ecosystem of open source, vendor-neutral projects

(CNCF via Github)

The Top-Performer Cloud Native Toolkit

Cloud native is critical because it allows you to leverage a variety of tools that take your organization's technology to the cutting edge. There is a vast landscape of programs associated with every phase of cloud-enabled DevOps. While the tools used to perform operations using a DevOps model are vital to execution, they are only a piece of the puzzle. However, it's worth noting that the technologies you'll need will fulfill a set of roles:

Microservice Architectures break down apps into bite-size pieces so that multiple team members can work on different pieces of a product at the same time.

Containerizing Traditional Apps means packaging up legacy programs so that they can function in modern environments and play-nice with other apps.

Containers & Kubernetes allow loosely coupled systems to work together seamlessly.

Continuous Integration & Deployment (CI/CD) bring testing and automation into your deployment processes so that you're never left waiting on manual, error-prone results.

Infrastructure-as-Code (IaC) Platforms ensure your systems are readily portable and adaptable by using declarative scripting.

Making DevOps Work For You

Getting Started

There is no one right way to implement DevOps. It means different things to different people and organizations. The best way to get started is to use DevOps principles to implement your DevOps practice.

A core DevOps principal is continuous improvement, as applied from lean manufacturing. However, before you improve something, you need to have something to improve. This is often referred to as the "Gemba walk" where you study an existing process with intent to improve it. So, start simple, observe your process in action, and start improving from there. (Citation: 2019 State of DevOps report)

First things first: trust, communication, and an improvement mindset.

Taking time to focus on the people and the culture is essential to high performance with DevOps.

Create Alignment

To eliminate silos between the phases of software development, effective communication is essential. DevOps team stand-ups, agile task boards, cross-functional teams, and defining mutual goals keeps everyone in the loop. A shared understanding and appreciation for your DevOps flow will promote a "one flow, one team" mentality.

Encourage Innovation

Allow developers to lead with their expertise – solving problems and experimenting along the way. Promoting their creative freedom to be innovative builds excitement around possibilities instead of the fear of failure. This includes letting each team choose their own development stack, as well as A/B testing with client features.

Spark Passion

Engage internal experts in DevOps, championing leaders to share their passions with others across teams. Putting heads together, though sometimes intimidating at first, allows for the passionate problem solving and team building collaboration.

Demonstrate Business Value

Measure DevOps progress in relation to your business goals to see its true value. Injecting metrics into the process allow teams to reach their goals faster and more reliably while demonstrating business value.

Guardrails

Safeguards should be baked into your process from the start to ensure there aren't bugs or security problems being delivered. While making mistakes during development is a learning process, guardrails keep the business safe from critical mistakes while still permitting the benefits of iterative learning.

Ok, but now what?

Like a New Year's Resolution, this all sounds great in theory, but can be challenging to put into practice. That being said, don't overwhelm yourself! Improving your teams' functionality doesn't start with a polished process out of the gate. It's a learning process that starts with manageable steps.

1. Begin with an agile backlog.

To keep things manageable, do not start with a tech focus. Do begin with an agile backlog with DevOps user stories. Write down your wish list for all the items on your plate and prioritize them appropriately into a to-do list. Once you have the backlog assembled, it becomes your continuously updated jump-off point for optimizing your DevOps proficiency and efficiency.

2. Build a capability matrix.

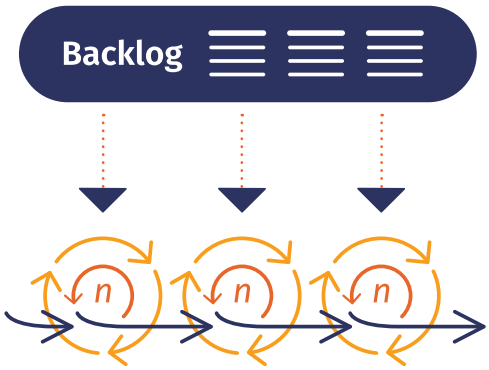
Identify capabilities within each DevOps area (app dev, pipeline, etc.), and assign user stories from the backlog to build the capabilities during each sprint. This matrix, like the backlog, should be constantly updated as another way of communicating what's in progress, as well as your level of DevOps maturity.

		Crawl			Walk			Run		
Area	Capability	S1	S2	S...	S1	S2	S...	S1	S2	S...
APP DEV	Local Containner Development	US1	US5	US...	US...	US...	US...	US...	US...	US...
	Test w/ Local K8s Cluster	US2	US7	U...	US...	US...	US...	US...	US...	US...
PIPELINE	Docker Build & Push	US3	US9	U...	US...	US...	US...	US...	US...	US...
	Deploy Brand to K8s	US5	US...	U...	US...	U...	US...	US...	US...	US
PLATFORM	IaC Create K8s Cluster	US6	US...	U...	US...	U...	US...	US...	US...	U
	Secure Application Ingress	US8	US...	U...	US...	U...	US...	U	US...	

Note: A full-scale capability matrix would contain dozens of capabilities.

3. Start small & grow with the Crawl-Walk-Run model.

The Crawl-Walk-Run model is built around two-week sprints, taking tasks from the backlog and producing tangible results. This is a plan of action on the path to a smarter DevOps that goes beyond planning and actually walks-the-walk.



Crawl

Establish a Thinnest Viable Product (TVP) flow through tech foundations, from app dev to deployment. Make your first sprint happen in two weeks to begin the process of conceptual alignment across teams.

Components: Pilot Containerized App, Simple Build Pipeline, Simple Script Deploy, and Dev Cluster

Walk

Now, complete the internal release of an application for real people to see. Examine attribution to lead time and, using your capability matrix, determine minimal requirements for internal release.

Components: Internal Release App, Multi-Branch Pipeline, and Test Cluster

Run

Release an application publicly to end-users, bringing the results of your hard work with DevOps to the outside world. This involves secure and resilient operation of your production platform.

Components: Production App, Secure Pipeline, and Production Cluster

Platform, pipeline, and app development must be a part of each sprint. **Repeat smaller crawl sprints, then walk sprints until you feel comfortable at each pace.** To gauge comfort levels, ask team members what they think: how do they feel about the sprints? This is an iterative process, so don't be afraid to keep pushing forward with new ideas, so long as each sprint results in operating code running in a cluster. Listen carefully, leverage team feedback and expertise, and soon enough, you'll find yourself running.

Be diligent about your backlog to keep track of your features and requirements wish-lists. Don't forget to track technical debts either. It's okay to take shortcuts to get something done, so long as you make sure to come back and fix it. Burndown of your backlog items is satisfying, but more importantly, can be used as a way of tracking progress and generating reports of real results.

> *"Stop making PowerPoints and write some damn code."*
>
> – Mark Panthofer, VP of DevOps & Cloud at nvisia

Remember, one size does not fit all.

The Crawl-Walk-Run model is a high-level approach that incorporates overarching best practices for DevOps implementation. However, when it comes to deeper, more complex models, there is no single, universal approach.

While there are many existing models (check out the book *Team Topologies*; see *Further Reading* for more), it's critical to determine what works best for your organization. Each organization has its own unique set of needs, goals, and constraints that evolve over time, so it's important to be flexible to these changes, too. Openness and agility will afford you long-term success.

Measuring Success

Success in DevOps is measured in numbers and a marked culture shift. Keeping track of key metrics makes success undeniably visible to team members and other interested parties within your organization, bringing a sense of pride to your teams and proven ROI for all interests.

Quantitative Key Metrics You Should Track

Lead Time: How long it takes for code to be running successfully in a production environment.

Deployment Frequency: How often code is deployed to production or released to end-users for your primary service or application.

Change Fails: The percentage of changes to production or released to end users result in impairment or outage and require a fix (i.e. rollback, patch, hotfix, etc).

Time to Restore: The amount of time it takes to restore service following an outage or service defect that affects users.

The State of DevOps Report, 2019)

There's more to DevOps success than plain numbers.

A cultural shift is felt, too. You'll witness your teams begin to trust each other more. By understanding everyone's part in a process and how it contributes to a whole, teams function as cohesive units instead of in siloed fragments. With frequent communication and feedback, knowledge of how a product works is universal, giving each team member a strong sense of ownership with results.

Confidence in themselves and the team allows for excitement toward trying new things and innovation. The technologies that come with DevOps are forward-thinking and propel your company's technology forward. With a learning culture that pushes continuous improvement, you'll be able solve old problems in new innovative ways.

Still Not Convinced?

A quick look at what the data has to say.

"Delivering software quickly, reliably, and safely is at the heart of technology transformation & organizational performance... including profitability, productivity, & customer satisfaction."

State of DevOps Report, 2019

The highest performers are two times as likely to meet or surpass organizational performance goals – odds that are in your favor. DevOps isn't just a new way of doing things; it's a method that carries business value and allows your company to respond at lightning speed to the ever-changing tech landscape.

If numbers speak to you more than words,

top performance with DevOps brings some astonishingly positive metrics. Per T*he State of Devops Report, 2019,* when comparing elite performers with low performers, the elite group has:

208x

more frequent code deployments

106x

faster lead time from commit to deploy

2604x

faster time to recover from incidents

7x

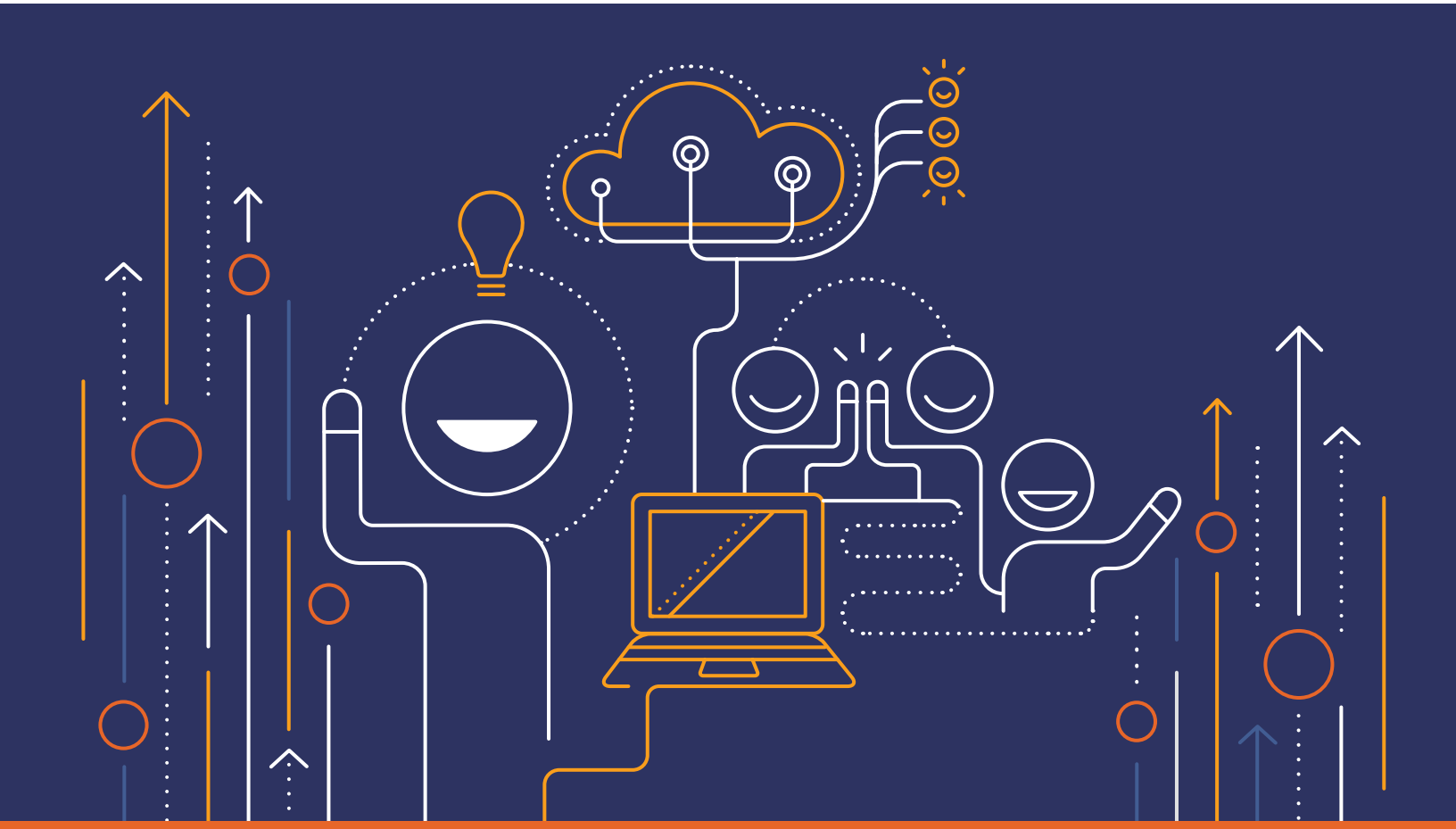
lower change failure rate (i.e 1/7 as likely to fail)

Beyond numbers, studies indicate that a combination of a responsive environment and perceived self-ability result in greater performance and self-confidence (*Druckman & Bjork*). By promoting a culture of flow, feedback, and continuous improvement that is reflected in technological processes, individuals feel like they are in an environment that supports them and values their success. Through learning, they gain valuable skills and improve their

performance, boosting confidence and willingness to innovate. This ultimately makes for happier employees and a healthier bottom line.

> ***"Industry velocity is increasing.*** *Many analysts are reporting the industry has 'crossed the chasm' with regards to DevOps and technology transformation, and our analysis this year confirms these observations. Industry velocity is increasing and speed and stability are both possible, with shifts to cloud technologies fueling this acceleration. This reaffirms the importance of technology that enables organization to deliver value to their stakeholders."*
>
> **State of DevOps Report, 2019**

In other words, **DevOps is critical to ensuring that your business can compete with and surpass others in your industry.** Falling behind now will cause concern in the short-term and stagnation in the long-term. By implementing smarter DevOps practices, your organization will see results reflective of those described above, prevent lethargy, and bring speed and optimized performance that provide a crucial competitive edge.



Your Smarter DevOps

Essential Action List

- **Assess where you're at** by examining key metrics. Pay attention to lead time, deployment frequency, change fails, and time to restore.
- **Create an agile backlog** of to-do and wish list items across your IT department. Look for opportunities for automation and increased communication.
- **Build your DevOps Capability Matrix** with user stories from the backlog.
- **Foster an environment** of communication and trust.
- **Define a TVP** for your first crawl sprint.
- **Complete a crawl sprint** that results in real code running in a cluster. Make it happen in less than two weeks.
- **Perform crawl and walk sprints** until you're ready to run.
- **Sprint, sprint, and sprint again,** pacing yourself while chipping away at the backlog.
- **Look back at your key metrics** to see how far you've come.

And with that, you've arrived at a smarter DevOps.

Have questions
or want to learn more?

The path to a smarter DevOps is attainable to all who set their mind to it, but made easier with experts who can guide you along the way. **nvisia** is proud to guide organizations, from small-scale to global enterprises, on that journey with tested knowledge and expertise.

Talk DevOps with nvisia.

Learn more at **nvisia.com.**

learn@nvisia.com

nvisia

Sources

nvisia Experts

Mark Panthofer
VP, DevOps & Cloud

Tim Liebl
Project Lead

Nick Schultz
Director

James Leiser
Sr. Technical Architect

Additional Sources

Deepti Suri

Director, Cloud Infrastructure DevOps, Automation, and DBA Services at Foot Locker

Cloud Native Definition v1.0

by CNCF, via GitHub

State of DevOps Report 2019

by DORA & Google Cloud

The Principles Underpinning DevOps

by Gene Kim

"Self-Confidence and Performance"

from Learning, Remembering, Believing: Enhancing Human Performance, pp. 170-175. Edited by Daniel Druckman and Robert A. Bjork, published by the National Research Council

Virus forced schools online, but many students didn't follow.

by Julie Watson and Carolyn Thompson, Associated Press for ABC News

Further Reading

The Phoenix Project

By Gene Kim, Kevin Behr, and George Spafford

A fiction novel that shows DevOps in action through stellar storytelling. It communicates the mindset and culture that surrounds DevOps.

The DevOps Handbook

By Gene Kim, Jez Humble, Patrick Debois, and John Willis

A non-fiction, dry equivalent of the Phoenix Project. It's a great toolkit but lacks a comprehensive framework to bring all the pieces together.

Measure What Matters

By John Doerr

This manual guides and informs how to bring DevOps into practice, helping set expectations, answer questions about what to do next, and how to measure results.

Team Topolgies: Organizing Business and Technology Teams for Fast Flow

By Matthew Skelton and Manuel Pais

IT consultants Skelton and Pais share an array of successful team patterns to help audiences choose the right DevOps topologies for their organization, keeping software healthy, teams happy, and business value growing as a result.

Thanks
for reading.

For more insights, visit nvisia.com/insights.