Gearset

Salesforce DevOps at enterprise scale

Discover the best practices and benefits of streamlining DevOps for Salesforce

Contents

Introduction

The best enterprise Salesforce teams are underpinned by great DevOps processes. The *2023 State of Salesforce DevOps* report has shown that **50%** of enterprise teams have adopted leading tools and processes, and of those teams **42%** are seeing monthly return on investment (ROI) of over **$50,000**.

The *2019 Accelerate State of DevOps* report tells us that only **9%** of elite teams adopted DevOps with a waterfall, "big bang" approach. There's good reason for this. Implementing DevOps is not a one-time project to check off your list. It's a journey that will require continual improvement and iteration.

As you set out, it's best to begin with a roadmap that provides a clear understanding of what DevOps for Salesforce means in an enterprise context, and clear objectives for your implementation. This ebook should function as an introduction to the key principles and best practices for enterprises seeking to implement DevOps for Salesforce.

50% of enterprise teams have adopted leading DevOps tools and processes

42% are seeing monthly return on investment (ROI) of over $50,000 following adoption

Who is this ebook for?

If you're a member of senior leadership (VP Technology, VP Salesforce, IT/Operations Director), you'll understand how and why DevOps for Salesforce contributes to the overall success of your company's strategy. You'll see that investing time and resources into streamlining your Salesforce delivery will reduce organizational risk and boost your Salesforce ROI.

If you're responsible for a Salesforce delivery team of more than 10 dedicated Admins, Developers, Architects, or Business Analysts, then this ebook is for you.

If you're a Salesforce Platform or Product Owner, then you'll gain useful insights into how to manage your org and build a successful culture for your declarative and programmatic configurators.

What is DevOps?

DevOps aims to streamline an organization's development workflows, using the best practices for processes and tooling.

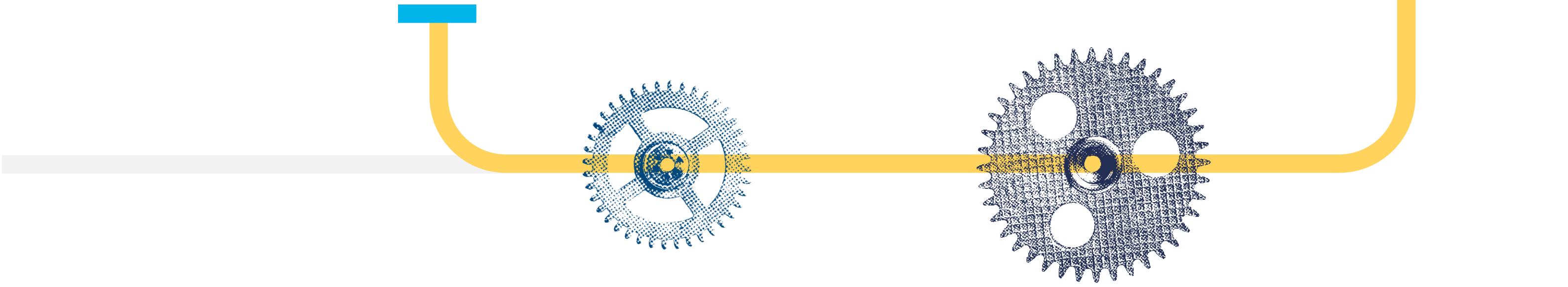
Though DevOps takes these processes from software development and IT operations, the fundamental problem that DevOps aims to solve is related to people. Specifically, DevOps is designed to break down the silos in a release process between developer and operations teams.

For example, in the world of Salesforce development, DevOps helps bring together every member of the Salesforce team to work harmoniously — with a shared knowledge, understanding of, and responsibility for, the delivery process.

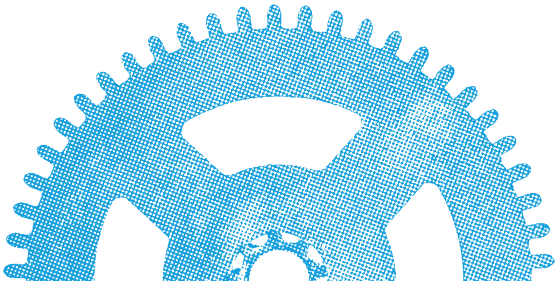
Why DevOps for Salesforce is different

As Salesforce is a Software-as-a-Service product, some major challenges of software development are handled for you. Traditional DevOps responsibilities such as hosting, infrastructure, scalability and security are — for the most part — already taken care of.

DevOps for Salesforce is concerned with the delivery process itself, and how teams collaborate effectively to deliver changes. The primary focuses are how Salesforce orgs are managed, and how changes reach production. This means that the whole Salesforce team can, and should, be involved in as much of the delivery process as is possible.



Salesforce is a unique platform, with unique metadata and custom code that doesn't mimic traditional software delivery. This makes Salesforce delivery challenging — perhaps surprisingly so given how easy it is to build on the platform. To seamlessly deliver robust applications and overcome these significant hurdles, Salesforce-specific solutions and continuous investment in your team are required.



The benefits of DevOps for Salesforce

A well-oiled and efficient DevOps process with reliably repeatable deployment practices has a whole host of benefits that save more than just time:

- ✓ Applications and system updates delivered to end users faster — the crux of digital transformation
- ✓ Tighter feedback loops between development teams and end users encouraging user-driven development
- ✓ Increased collaboration between all stakeholders
- ✓ Decreased organizational risk
- ✓ Increased resilience in the event of data loss or corruption

There are even more benefits besides, but they all boil down to one key outcome:

more value, more often.

These benefits clearly translate to a Salesforce context. A poor DevOps process hampers the ability to get the full value from Salesforce, which impacts the wider business as Salesforce is a mission-critical part of your business' technology stack. In fact, 98% of teams report that Salesforce is integral to meeting wider business objectives. Letting new, business-critical applications, features, and developments stay stuck in the development cycle as a result of poor process will drain your ROI as it creates:

- ✕ Less efficient Salesforce end users
- ✕ Hampered efficiency of Salesforce delivery teams
- ✕ Paid-for functionality that's obsolete until utilized
- ✕ Employee dissatisfaction, which causes high turnover

Why DevOps for Salesforce is needed

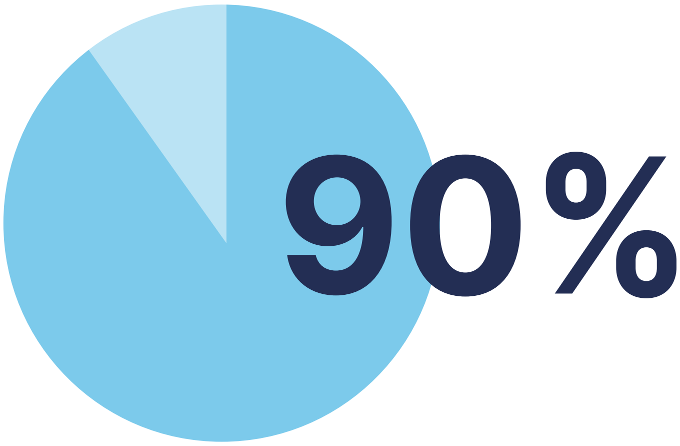
Salesforce implementations are getting increasingly complex. The 2023 State of Salesforce DevOps report found that 90% of organizations had increased demand on their Salesforce team. 85% of organizations also reported that their dependence on Salesforce had increased.

With more new products, more changes, and more intricacies to Salesforce offerings, it's becoming harder for Salesforce professionals to keep up to date and develop detailed knowledge. This presents a larger challenge to enterprises, who are operating on a greater scale. A mature and well-oiled DevOps process alleviates a significant amount of pressure on this team, and frees them up to spend more dedicated time learning how to get the most out of the Salesforce platform to benefit the business.

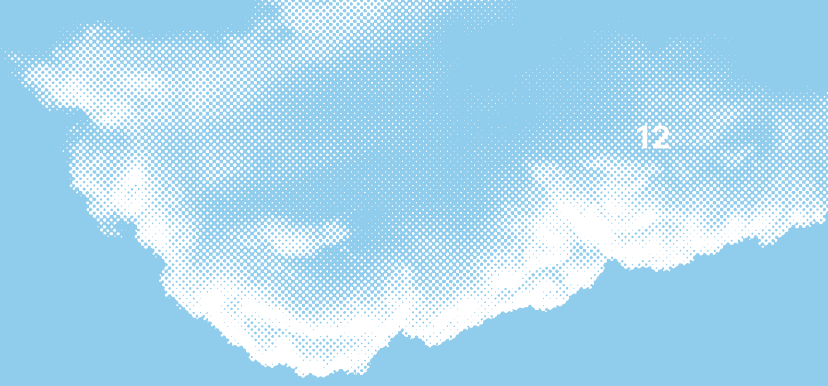
Leaders aren't just concerned with the complexity of the Salesforce platform. They're also being forced to rapidly adapt their business strategy in line with changing markets and the fluctuating state of the global economy.

With Salesforce as a cornerstone of your business strategy, teams need to have the agility to pivot, and fast. Delivery methods can hamper this ability to pivot or bring new products to market, especially as customer attention spans are shorter than ever. If you're not there to capitalize when it matters, then you'll lose to the competition.

Salesforce is a significant investment for your business. Streamlining your Salesforce DevOps process is one of the most effective ways to see increased returns on your investment. With DevOps you give yourself the best chance to prevent efficiency sinks, free up your team to research and implement new platform capabilities, and empower the business to pivot, meeting the market where it is, when it is.



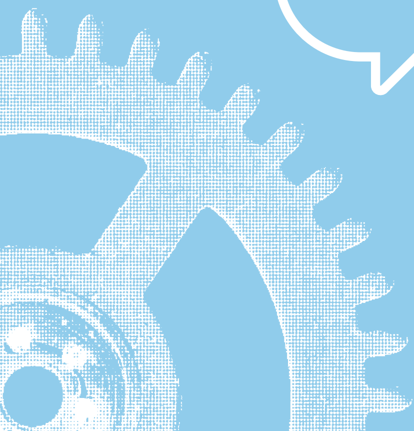
The 2023 State of Salesforce DevOps report found that 90% of organizations had increased demand on their Salesforce team.



The components of a **successful** DevOps process

There are many moving parts in your DevOps process, technical and non-technical. While both are important, neither can be truly optimal without the existence of the other.

In this section, we'll cover the steps to establishing a successful process with context for enterprise organizations. It's worth noting that enterprises face specific and amplified challenges. At scale, it takes more effort to break down silos and manage dissenting opinions on technical decisions.



Culture

A strong DevOps culture forms the foundation of any successful DevOps process. It's also the hardest thing to implement if the desire for change isn't there, and a challenge to maintain if there isn't mutual understanding of the reasonings behind your DevOps processes.

One of the key benefits of DevOps is breaking down silos between teams and individuals, which requires a level of communication and transparency that can be complex in an enterprise organization. While it may be uncomfortable to open up work to judgment and scrutiny, leaders can mitigate this by upholding the value of peer review. Transparency creates opportunities for feedback and learning from different angles, and will contribute to individual personal growth.

DevOps enables continuous improvement and tight feedback loops, which can have numerous benefits, including being more successful and more fulfilled.

Strategy

Businesses make significant investments in their Salesforce orgs, and expect to see returns. Therefore, it’s essential to have a solid strategy in place for delivering changes and new applications to your end users across the orgs that you manage. And this strategy will and should be expected to change over time. The key principle of DevOps is continuous improvement, and as such your strategy and delivery process should follow this philosophy.

You don’t need to reinvent the wheel when it comes to crafting a DevOps process that works for you. But there are some key areas that you should consider:

→ Communication

Clear and efficient lines of communication should be established between all the stakeholders involved. Configurators, Testers, Business Analysts, Users, and Leadership should all have visibility into the development cycle, and lines of communication that establish trust. We recommend setting up automatic progress notifications, via services such as Slack, which will be delivered to the relevant stakeholders.

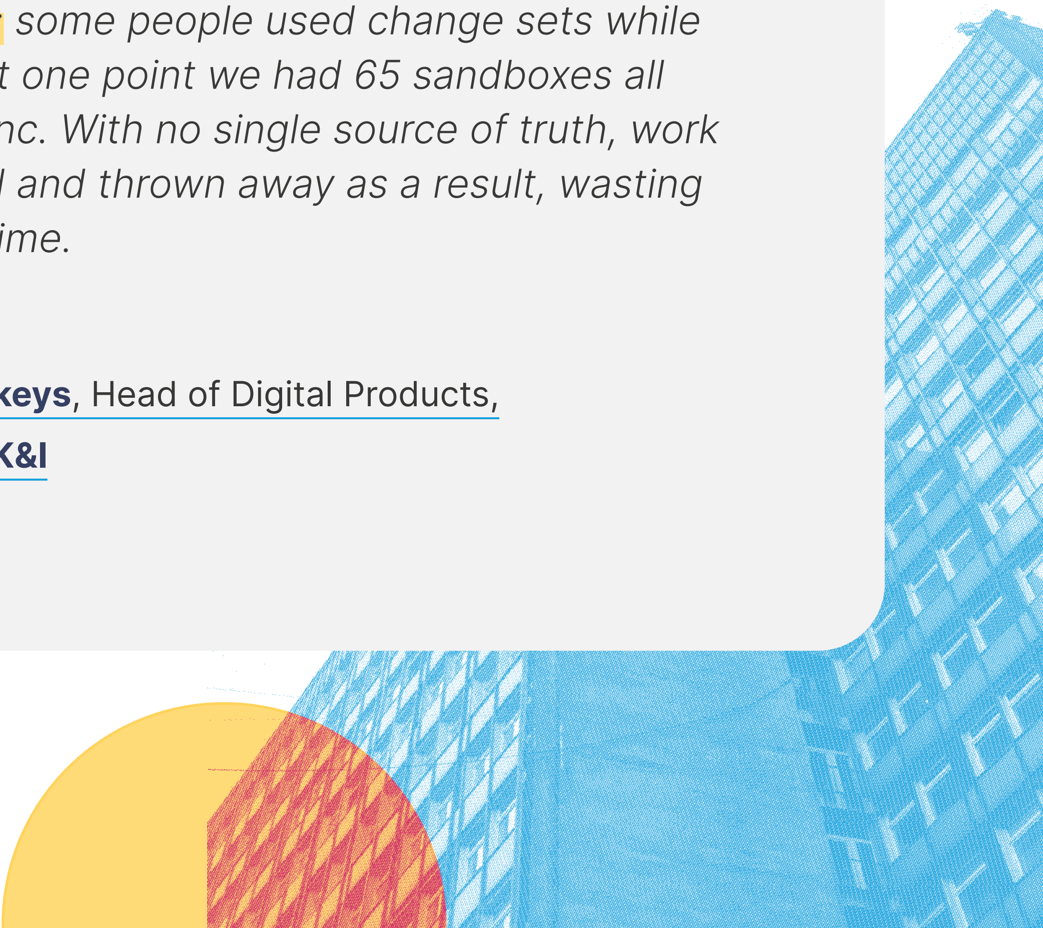
Communication isn’t just about the methods used. It can only be effective when it’s clear and concise. And this is incredibly important in an iterative DevOps environment.

→ Application management

You're likely already using an IT service management system like JIRA, ServiceNow, or something similar to manage user stories and epics for your development. Your strategy shouldn't only consider how these applications are used, but how they're integrated into your development lifecycle for greater transparency and vision. With transparency and visibility into what each member of the team is working on, it makes it easier to distribute workload, anticipate delivery windows, and make sure work isn't being duplicated between individuals or teams.

> *"[Before adopting DevOps] different teams and individuals worked in silos. This meant it was difficult for us to see who was working on what. And there was no consistent deployment process: some people used change sets while others used SFDX. At one point we had 65 sandboxes all completely out of sync. With no single source of truth, work was often duplicated and thrown away as a result, wasting precious developer time.*

Paul Watkeys, Head of Digital Products, **Veolia, UK&I**



Version control

A Git-based version control system (VCS) is at the heart of any modern DevOps process. Version control allows your team to track all the changes made and easily review individual contributions, while making it possible for multiple contributors to work on the same features at the same time.

Individual developers and configurators contribute new work within Git branches, which are later merged into the main branch — containing the latest and stable version of the project or org — once the work has been reviewed.

There are countless benefits of using a version control system like Git. Among many other things, it helps you:

- Reduce risk and avoid costly mistakes, by making Git the source of truth for your development team instead of your production org.
- Track and annotate your development work with a complete history of every change your team makes to your org — to help your auditing and debugging processes.
- Collaborate with multiple team members working in the same org without overwriting each other's work or treading on each other's toes.
- Release reliably and safely by testing your changes first until you're satisfied they're fit for purpose.

→ Branching strategies

The biggest consideration in your use of version control is going to be the branching strategy.

Your branching strategy will dictate how your team interacts with the various environments in your pipeline, and forms the basis for how your automated pipeline will work. The branching strategy most commonly used by large teams is an “environment” branching strategy, i.e. orgs backed by persistent branches.

The C2FO team also had version control on their radar, as they wanted to improve collaboration and introduce code reviews to their process.

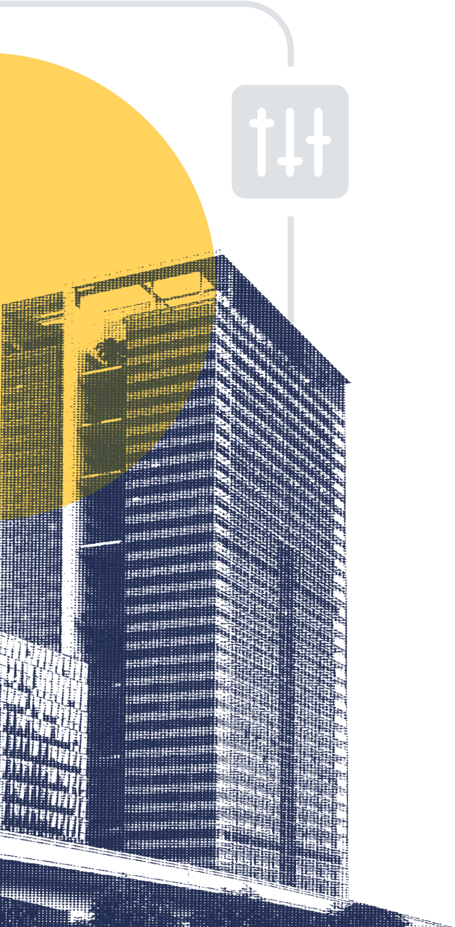
> “
>
> *We needed to build our release process, and we also needed version control — so we were looking for a solution that could do it all.*
>
> **Corrine Walker**, Salesforce Engineering Manager, **C2FO**

More information on branching strategies, and how to assess what might work best for you can be found in our Version control for Salesforce whitepaper.

Testing

Unit testing is an integral part of software development, reflected in Salesforce's requirement that developers test at least 75% of any new code they deploy to a production org. But the tests that are written need to be useful and meaningful to make sure your team can easily review and reliably build on your code in future.

Automation can help you by testing changes every time they are merged into the main branch and/or deployed to another environment. An agile DevOps setup will also let you populate development or testing environments with data from production, so you can check your code works with actual data — real (albeit sometimes "dummy") data that will come in all sorts of shapes that you might not have thought about when you started writing your code.



Automated testing allows you to:

→ Save time and effort, by making sure your changes are deployable.

→ Avoid shipping faulty code and causing existing functionality to break.

→ Make sure that you're developing the precise features that your organization needs.

→ Improve the quality of your development work by reducing bugs and critical issues, freeing up your team to work on those great new features.

→ User Acceptance Testing

As part of your testing strategy, you must have a level of user acceptance testing (UAT) — this ensures that the features and changes you've made meet the user requirements from the story and perform the desired function.

Build out a mechanism where "power-users" from each area of the business that Salesforce touches (Sales, Service, etc.) are involved in the DevOps process and are able to give feedback on new features. This is critical to ensure configurators and developers are building solutions that help address the real needs of the business.

> "*I don't have to spend much time reviewing the work or worry that, even after two days of reviewing, it's going to bomb. I get peace of mind knowing the deployment will work.*
>
> **Vijay Aswani,** Salesforce Technical Lead, **UCLA**

CI/CD

Continuous integration (CI) builds on a Git-based workflow and automates the process of testing and validating changes, making sure that they can be deployed. Continuous delivery (CD) is about releasing frequent, small changes to users via an automated process and reducing the risks associated with big releases.

Taken together, CI/CD takes the pain and risk out of deploying changes to multiple development environments and on to production. It promotes an iterative and incremental approach to development, where you aim to contribute frequent small changes and get immediate feedback from your team and end users.

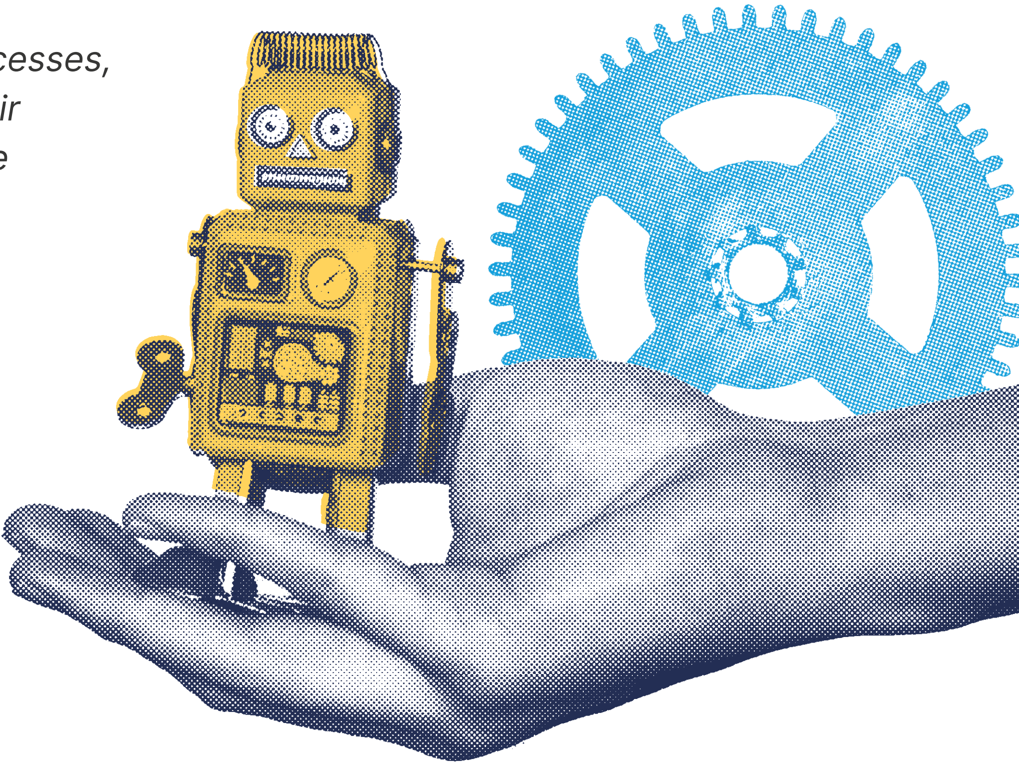
Using CI/CD to make regular, small and automated releases helps you:

- ✓ Avoid manual errors causing problems with releases.
- ✓ Reduce the risk of individual releases as your process becomes ever more reliable.
- ✓ Reduce downtime by quickly rolling back mistakes and unwanted changes.
- ✓ Tighten the feedback loop and allow your users to drive your development process.
- ✓ Work on multiple projects simultaneously and change direction in response to feedback without losing the existing work.
- ✓ Release the value you’ve created to your users as fast as possible.

It’s rare for an enterprise-sized company to adopt continuous deployment, i.e. automated releases to production, due to compliance requirements and risk management. However, your process can get to a stage where it’s automated until the very last click of a button so that your releases are near seamless.

> *“Their [Salesforce Delivery] team decided to create several CI jobs for each of the five levels of their deployment process. With this new setup, issues can be caught earlier and isolated to upstream environments, rather than reaching UAT and blocking other work. Adopting [DevOps tooling] really started an evolution, taking us to where we are today.*
>
> **Sr. IT Analyst**, Global Goods Manufacturer

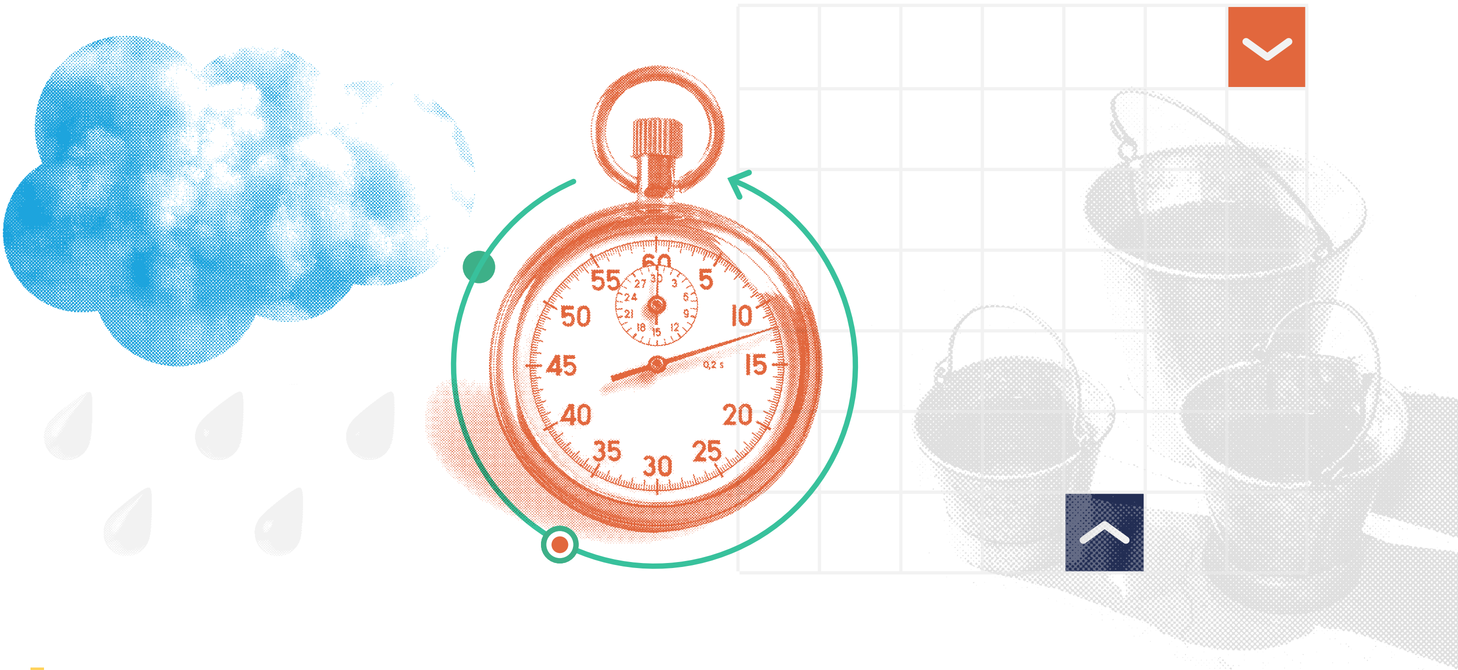
An analysis of CI/CD processes, and how to maximize their effectiveness, is available in Gearset’s free ebook CI/CD for Salesforce.



82%

of Salesforce teams are working towards CI/CD

Source: State of Salesforce DevOps 2023



Backup and recovery

Backup and recovery is key to mitigating organizational risk, and the reputational damage that has the potential to arise from a data incident.

A backup of your customer and organization's data is essential in case a disaster strikes — whether it's accidental data loss, data corruption due to a third-party integration, malicious actions of disgruntled employees, or even a Salesforce platform outage.

But a backup solution is only as good as your ability to restore from it. Backups and org security are fundamental to DevOps, as it ensures the business can quickly resume normal operations.

Key here is the relationship between your data and metadata, the latter storing your data and providing the structure of your org. Many teams only consider the need to back up their data, but it's critical to protect your metadata too. Teams that have tools and processes in place to monitor and roll back changes to their data and metadata are then also in a position to restore complex data hierarchies to their orgs quickly, and before too much damage is done.

A smart backup solution will allow you to:

- Monitor your org's data for changes or deletions along with metadata monitoring
- Use your familiar deployment process to restore data and metadata
- Deploy backup data to sandbox environments for testing
- Mask and delete records to comply with data protection regulations (CCPA, HIPAA, GDPR)

> *If we had a complete failure, I'm confident our data is safe in our Gearset backups and we'd be able to restore. It's like insurance — you need it, but hope you never need to use it.*
>
> **Chris Deutschmann**, Configuration Consultant, **Sage**

For a detailed guide to backing up and restoring your Salesforce org, read Gearset's free ebook Backups for Salesforce.

High-performing DevOps teams typically restore service in under an hour

Source: Backups for Salesforce ebook

Charting your Salesforce DevOps journey

A quality Salesforce delivery model relies on a sound understanding of the methods for delivering change. How a large organization goes about handling both day-to-day changes and long-term changes influences how successful they are in the long run. This is a core responsibility of a Center of Excellence.

Establishing a Salesforce Center of Excellence (often referred to as a CoE) should be a top priority. In fact, a *10k Advisors* report found that **91%** of the most successful companies seeing max ROI on Salesforce have a CoE established.

DevOps falls into the responsibility of these core responsibilities of a CoE:

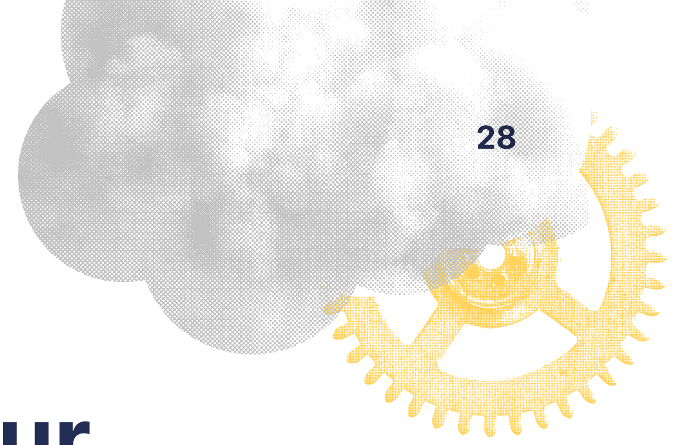
→ Leadership
→ Governance
→ Change management
→ Tooling
→ Standards
→ Metadata management

You'll bring the best DevOps practices to the fore by establishing an effective CoE with a holistic view of Salesforce as a platform. This should include a strategy for your DevOps implementation and a roadmap for how the CoE will act as a force for change.

A Center of Excellence also brings in stakeholders from across the business. For example, vision and leadership could be the responsibility of C-level executives, configurators, and/or end users. Establishing a CoE that includes a variety of stakeholders promotes a business-wide understanding of your DevOps processes, tooling, and objectives for your use of Salesforce.

When it comes to change management, you'll generally need to slow down in order to speed up. Take time to understand the Salesforce org(s) that you look after, understand the long-term strategy of CRM in your company, and build a CoE with roles and responsibilities that support it. Continuous improvement underpins DevOps practices and philosophy, and your CoE should always be thinking about ways to continuously improve and make sure that Salesforce both stays operationally relevant and can drive business growth.

It's inevitable that some employees will be resistant to change, which you'll have to accept. But, with coherent strategy and clear communication, disruption will be minimized.



Who to consider when building your DevOps process

Though your DevOps process impacts everyone at your organization, there are three main groups that you should consider when determining your approach.

1 The wider business

Your use of Salesforce should allow you to meet business-critical objectives. The Salesforce platform has the ability to take an organization to atmospheric heights, and your DevOps process is critical to that.

Businesses should be looking to serve customers consistently with more innovative and satisfying solutions — Salesforce undoubtedly has the capability to do that. However, the ever-changing commercial landscape makes it exceptionally difficult to stay on top of consumer trends. The ability to collect, process and store relevant data can allow you to make sound business decisions that drive commercial success.

The customer experience should be at the forefront of any great Salesforce implementation, as your business objectives ultimately serve these customers. The faster and more efficiently you can serve them with a relevant, coherent service and products that meet their needs, the better. If you're relying on Salesforce, then you're relying on your DevOps process to meet customers where they are, when they are there.

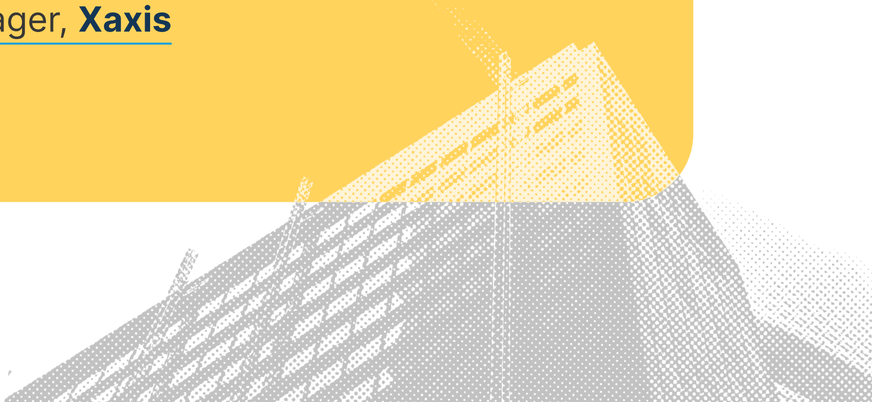
2 End users

As a leader in a Salesforce team, or an executive team responsible for Salesforce, it’s your responsibility to be able to communicate why and how Salesforce is changing in your organization.

As the complexity of your org increases, and with it the velocity and number of changes, your end users must be included and well informed to ensure the success of the features and functionality being delivered. Make sure that you’re not building towards an unattainable utopia. Build things that are useful and that are driven by constant, reliable user feedback.

DevOps plays a key role in being able to drive this level of engagement and change. A great DevOps process supports high levels of change velocity. This keeps end users engaged, and fosters belief that they are being supported in their operational goals by the Salesforce development team. This is hugely important to overall business success.

> “
>
> *Previously, we had issues holding on to development work too long which made deployments even more complicated. Now, we’ve been able to build a better relationship with our end users, which is a big win.*
>
> **Alex Jones**, Project Manager, **Xaxis**



3 Salesforce team

Your Salesforce team needs the skills to deliver this level of change. While some of these skills will exist, some team members may be less familiar with new methods and technologies. Failure to address this skills gap limits the success of your DevOps implementation and overall use of Salesforce. 41% of Salesforce teams report that their team lacking experience is a restricting factor for managing their Salesforce releases.

Start with an evaluation of existing skills in the team. Moving personnel into roles that are best suited to the business' needs, and that meet their personal goals, is a great way to drive success of your process and, holistically, your organization. Pulling together and comparing this information will take time and effort. But, through planning and assessing these two things together in detail, you'll drive employee satisfaction, engagement, and ultimately the growth and success of your Salesforce strategy.

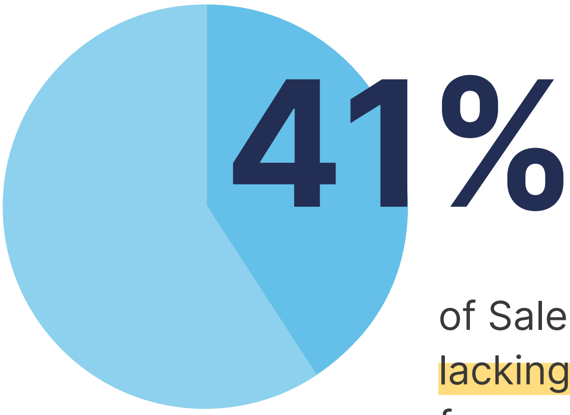
That being said, time needs to be set aside for learning and personal development. Investing in your process is all well and good, but neglecting the learning journey of your employees that are primarily responsible for delivery of Salesforce will hamper the effectiveness and ROI of your investment.

Fortunately, plenty of resources are available to get your team started.

Salesforce's Trailhead is a great learning platform for technical and soft skills. For learning dedicated to DevOps, you can take advantage of free resources such as DevOps Launchpad.

You'll also find value in your team attending conferences hosted by Salesforce and the Salesforce community, where they can network with teams solving similar challenges, and learn from experts in their field. DevOps Dreamin' is a fantastic example of a community conference that builds specific knowledge and expertise.

There are no hard-and-fast rules for your DevOps journey. It will develop as your processes, people, and technology matures. But having a vision of where you want to get to, and the resources at your disposal to get there, you'll be able to drive towards ongoing success.



41%

of Salesforce teams report that their team lacking experience is a restricting factor for managing their Salesforce releases.

Source: State of Salesforce DevOps 2023

Measuring success

It’s critical to measure the success of your DevOps processes, because:

→ It allows your CoE to understand what is/isn’t working

→ It encourages the visibility, collaboration and feedback that underpins a successful DevOps culture

→ You can quantify deliverables to the three key stakeholder groups (detailed above)

→ You can continuously improve

The *2023 State of Salesforce DevOps* report shows that teams with the highest performance metrics have mature DevOps processes in place.

Overall, 98% of Salesforce teams see ROI from implementing a DevOps process, with 27% seeing over $50,000 worth of savings a month, without taking into account the nuances and business acceleration achieved through digital transformation.

So what does the success and the impact of a great DevOps process look like?

Process

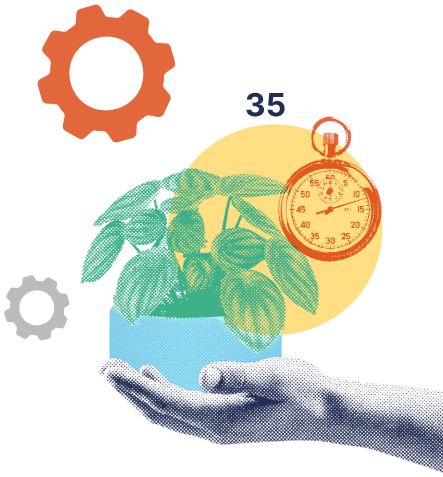
The DevOps Research and Assessment (DORA) metrics are the key metrics that help us to identify the efficiency of our DevOps process. Along with examples, which are considered "high performing", they are:

Daily
Deployment frequency

Less than 10%
Change failure rate

Less than a day
Lead time for change
(between work being completed and released)

One day
Mean time to recover
(data/metadata issues)



→ People impact

Optimizing your processes and aspiring to trailblazing DORA metrics is important. However, striving to do that needs to be balanced by the human factors that contribute to those numbers.

For example, if any of these metrics indicate high performance but only as a result of your staff regularly working 12+ hour days at their desks, you're unlikely to have a happy team. The *2023 State of Salesforce DevOps* report showed that teams are spending an average of 9 hours each month deploying outside of their contracted hours. This is not sustainable.

While having an optimal process is admirable, getting there should come from iterating and improving with the resources available within the team.

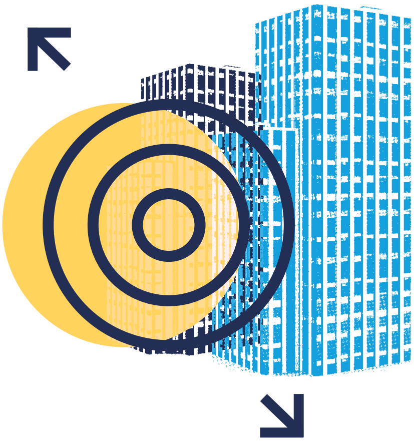
Being mindful of the following, while continually assessing the business requirements and goals, will enable you to identify where to hire additional resource, address technical issues, or tweak processes:

- Ability to collaborate
- Employee sentiment
- Staff retention
- Amount of overtime required
- Non-specialist consultant spend

These parts of your team's culture *will* have a knock-on effect on how successful changes to your processes and delivery will be, but they aren't mutually exclusive.

As markets and political climates change, many businesses will be inclined to remain in situations that are sometimes suboptimal. It's incumbent on leaders to make sure they create and foster a culture that doesn't perpetuate negative behaviors at the expense of staff.

The *2023 State of Salesforce DevOps* report showed that teams are spending an average of 9 hours each month deploying outside of their contracted hours.



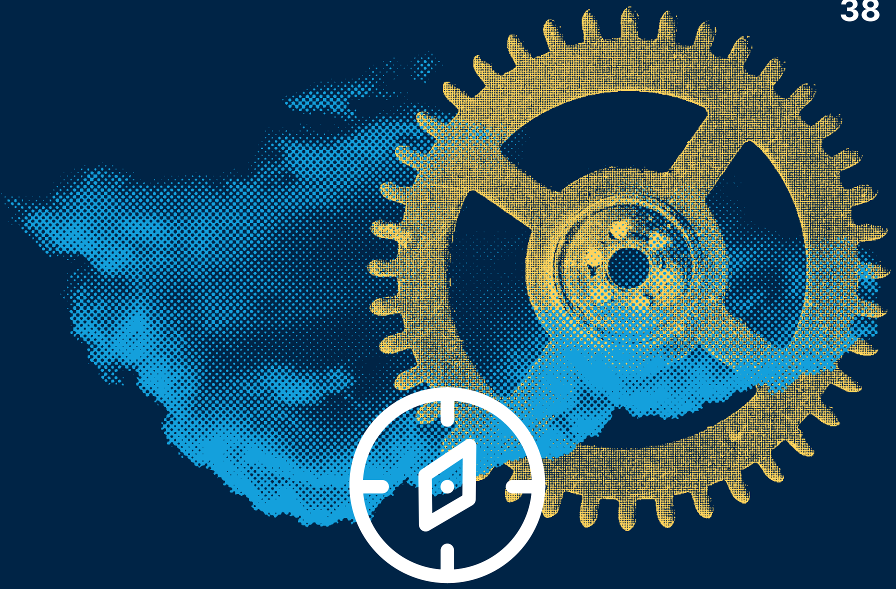
→ Business impact

DevOps has a direct impact on the overall ROI of the Salesforce platform and the overall digital transformation strategy. While the DORA metrics can give you a great understanding of your DevOps process' effectiveness, this view is too narrow.

Leaders and executives are obsessed with customers and making them happy. Salesforce touches a large portion of an enterprises' customer base. Therefore, effectively managing the platform will directly impact the overall performance of the business.

Providing the right things are built for the right people (which could be a whole other ebook in itself), you'll see the following impact:

- ✓ Faster digital transformation of services
- ✓ Increased revenue
- ✓ Longer customer retention
- ✓ Positive brand awareness
- ✓ Higher customer satisfaction (NPS etc.)



Building a DevOps Solution

The path to maturing your DevOps process involves decisions as to what solutions you should adopt. Once you've considered strategy, culture, and formulated an idea of the journey you'll go on, it's time to **think about the tech you'll use to help you**.

Again, when it comes to solutioning your Salesforce DevOps process you'll need to consider both:

- Your existing technology stack
- The skills of your Salesforce team

Alongside these two concerns, there are plenty of considerations and a number of ways you can make sure you pick the **most efficient solution that will deliver ROI and scale along with your business.**

The business case

To justify the spend on DevOps solutions, you need to begin with a clear understanding of why you're investing in revamping your DevOps process, which has already been well covered in this ebook. Any proposal for change needs to be supported by a business case, especially when budget is being requested, spent, or there's interruption to BAU activities — which can often be the reality of making changes to the DevOps process.

A compelling business case for any Salesforce-related technology, people, or process change needs to demonstrate a positive impact on at least two of the following:

- Return on investment
- Skills and knowledge
- Business system user satisfaction
- Team happiness and/or productivity

Fortunately, DevOps directly or indirectly impacts all four of these areas. There are also additional benefits that often mean that the ROI of improving your DevOps process is significantly more.

The time saved by each member of your team could not only be put towards backlog items, new tickets, or CPD. Those employees could also learn more about the Salesforce platform and help leverage it to drive the commercial growth of your business.

The possibilities are endless — with more time in your employees' hands, and a platform that optimizes digital transformation.

Tooling

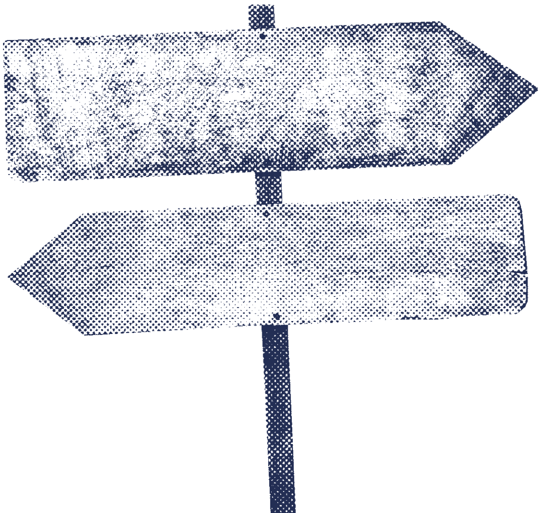
Technology plays a huge part in the ease of adoption, scalability, and value derived from your DevOps process. There are two key ways that large teams approach tooling. They may decide to build their own custom solution, or they work with a vendor to purchase a Salesforce DevOps platform. Each has benefits and drawbacks, but there tends to be a preference for working with vendors.

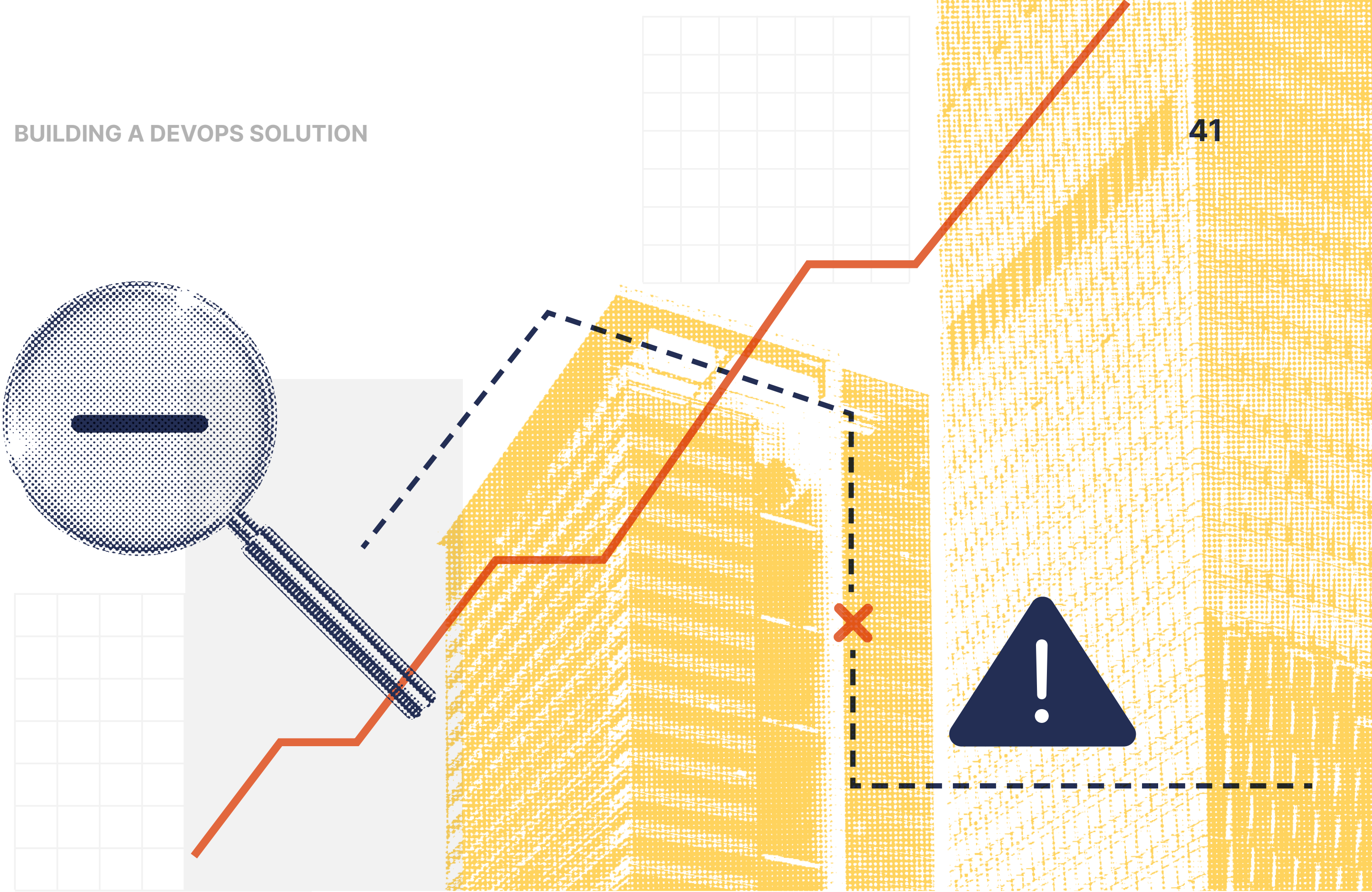
→ Build your own

Many large organizations with an existing software development team might initially see building an in-house DevOps solution as a shortcut to implementing DevOps for Salesforce — especially when there's the opportunity to use parts of an existing tech stack.

For many organizations, building a platform in-house can be seen as a solution to retain ownership of security across new processes. It can also be a route to maintain control of new technical solutions and add customized features, rather than introducing third-party software.

However, it's common to see businesses underestimating the complexity of the Salesforce platform, which has a number of unique features. Many IT departments have embarked on building DIY solutions only to discover that this approach ultimately fails to deliver on the promise of DevOps.





Common drawbacks of building a bespoke solution are:

- ☒ It’s slow and takes significant resource
- ☒ Susceptible to uncapped rising costs
- ☒ Increased organizational risk from single-points of failure
- ☒ Siloed knowledge
- ☒ Significant ongoing maintenance efforts

These issues will be experienced when building your own solution for any technology stack, but Salesforce brings its own unique challenges that exacerbate them.

Salesforce makes multiple changes and releases a year that render the maintenance of a bespoke solution constant. Should a member of the team leave who has partially built the process or solution, that knowledge leaves with them and your process could grind to a halt.

→ Vendors

Even with a multi-skilled software development team, it’s clear that building a Salesforce DevOps solution in-house is a resource-heavy process that requires domain expertise and raises many key challenges.

Thousands of large Salesforce teams are choosing to implement a purpose-built DevOps solution to meet their business objectives, bringing with it a range of benefits such as:

- → Finer control on costs and team resources
- → Solves challenging Salesforce-specific nuances
- → End-to-end solutioning
- → Out-of-the box easy to use and scalability
- → Expert technical support

It’s important to understand that when you purchase a Salesforce DevOps solution from a vendor, there are hundreds of thousands of hours that have gone into solving the specific Salesforce challenges involved. As a result, you’re not just buying the technical solution — you’re buying quality services, domain expertise, and deep knowledge that, without hiring directly from those talent pools, you won’t find elsewhere.

→ If you decide to buy

If you do decide to purchase a solution, there are some key areas you'll want to delve into when working with a vendor. You'll also want to involve the deep subject matter experts in the team, that have pain points, to help you evaluate. And always, always ask for a trial.

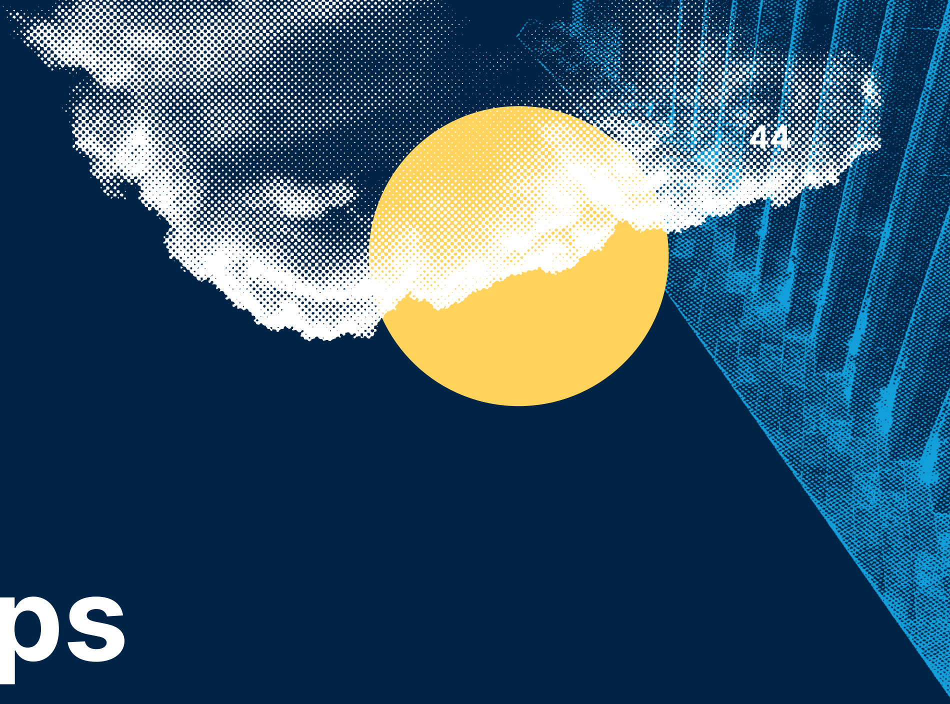
Key areas to ask a vendor about when engaged in a sales cycle:

- Salesforce domain expertise and insight
- Ease of onboarding and low-code capability
- Access to expert DevOps support
- Proactive maintenance

Equip yourself with a suite of questions and evaluate these areas, but also ensure you keep your specific requirements front of mind.

While the challenges that vendors solve are largely universal, make sure you ask questions that relate to your organization's use of Salesforce products, alongside concerns with governance, security, and team knowledge gaps that aren't currently supported in-house.

For in-depth detail on this paradox, you can find more info in Gearset's Build or Buy whitepaper.



Next steps

If you’re ready to see how investing further in your DevOps process can streamline delivery of Salesforce in your organization, arrange a consultation with us. We’ll give you specific, tailored advice to help you reach your business objectives.

Book a consultation

If you want to learn more about DevOps then visit DevOps Launchpad for free training and certification. Join the thousands of teams leveling up to accelerate the growth of their companies.

Start the learning journey

About Gearset

Gearset is the leading Salesforce DevOps platform, with powerful solutions for metadata and CPQ deployments, CI/CD, automated testing, sandbox seeding and backups. It helps Salesforce teams apply DevOps best practices to their development and release process, so they can rapidly and securely deliver higher-quality projects.

Thousands of Salesforce professionals use Gearset, and have shipped millions of deployments, run billions of automated tests, and backed up billions of records. With inbuilt intelligence that solves the fundamental challenges of Salesforce DevOps, Gearset is a uniquely reliable solution trusted by more than 2000 companies, including McKesson, Accenture and IBM.

W: gearset.com
E: team@gearset.com
T: +1 (833) 441 7687