

WHITEPAPER

# Capturing the business value of Compliant Database DevOps

By Kendra Little

DevOps Advocate at Redgate



# Capturing the business value of Compliant Database DevOps

## Contents

IT leaders today must increase business value, not simply focus on ROI	3
Why can't we scale value by adding more engineers when a project gets behind?	5
Lenses for examining the business value of DevOps	6
The new ways in which IT specialists and software engineers create business value	7
Global metrics for DevOps performance	8
Redgate's framework to iteratively increase value with Compliant Database DevOps	9
Summary	14
Learn more	15
Acknowledgements	15
Bibliography	16

# IT leaders today must increase business value, not simply focus on ROI

Information technology departments around the world are undergoing a revolution. As business owners realize that software core to business operations tilts the playing field in favor of those who use it most effectively<sup>1</sup>, they increasingly demand that IT leaders reinvent the way their departments work with all areas of the business.

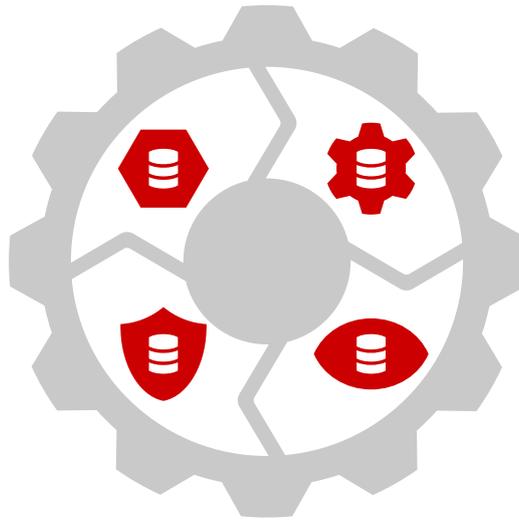
The demand to integrate information technology into business processes in new ways has increased dramatically in the last three years. In 2017, Gartner found that 47% of CEOs reported experiencing pressure from their boards of directors to transform their digital business<sup>2</sup>. In 2018, they found that 62% of CEO respondents had a management initiative or program underway to “make their companies more digital” – and this number rose to 82% in 2019.<sup>3</sup>

These changes have required IT leaders to rethink their approach to investment. In the past, IT departments have often been treated as cost centers which only indirectly contribute to revenue generation. The old concept of an IT department is of a group who provides a service to the business, fulfilling requests rather than acting as full collaborators with their colleagues in other business groups. Leaders in the older model of IT tend to invest in technology to improve the department’s ability to fulfill requests from a “cost savings” perspective: if we purchase X, how many human hours of labor might that save us, and therefore what is the return on our investment?

1. [Information Technology and Industry Concentration](#), James Bessen, 2017

2. [2017 CEO Survey](#), Kasey Panetta, Gartner

3. [Gartner Survey Reveals That CEO Priorities Are Slowly Shifting to Meet Rising Growth Challenges](#), Gartner, 2019



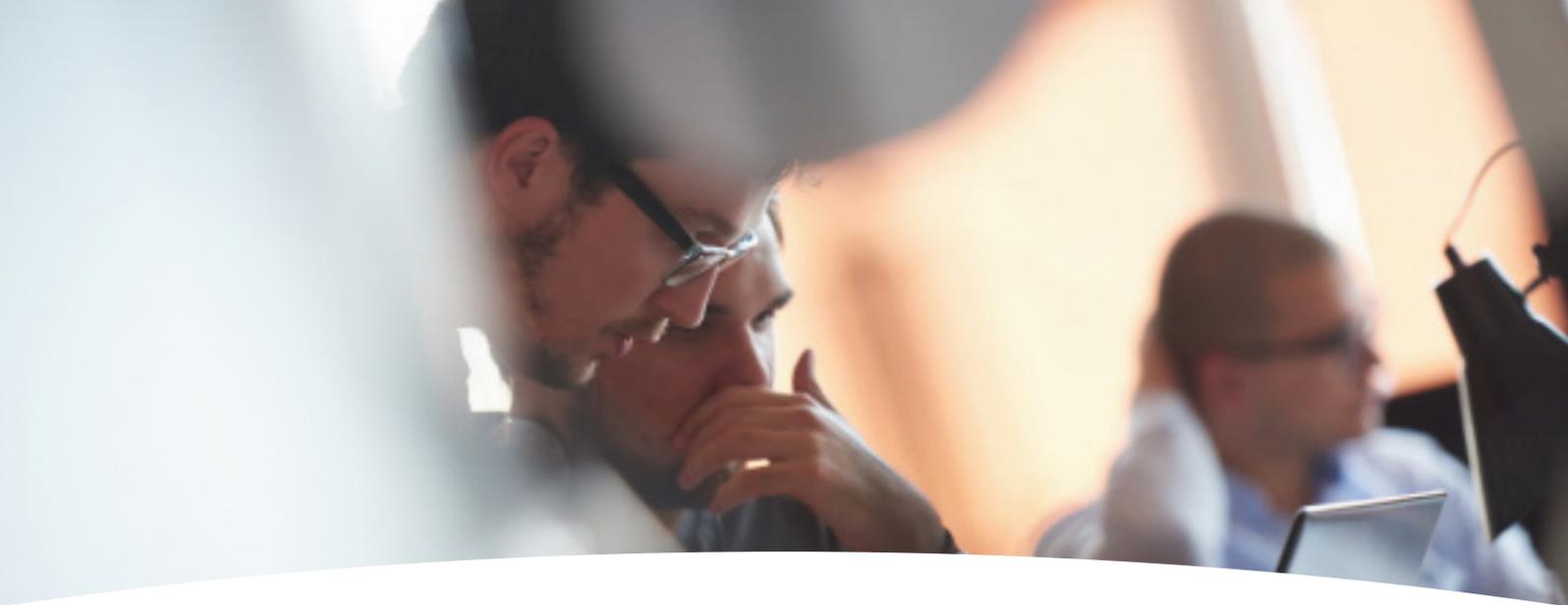
As IT departments change from the older cost center model into a collaborative group of engineers who work with business users and customers to continuously innovate and deliver new products and services, IT leaders need to think more broadly than cost savings and ROI calculations. Investments must align with strategies that increase the value which the company can deliver to its customers.

Put another way, IT leaders today must transform their digital capability. This requires them to identify solutions which help them meet and exceed their organizational goals year after year. This is a more difficult challenge than simply cutting costs, as it requires transforming the culture of your IT team and their ways of working with everyone across the business. But it also has the potential to yield much higher rewards.

The goal of DevOps is to enable the continuous delivery of value to end users<sup>4</sup>. This is identical to the goal of new IT organizations: engineers collaborate with others across the business to constantly drive and improve organizational performance. To enable this goal, we need to measure shared outcomes across software developers, IT staff, and users across various business silos.

In this paper, we will first discuss why scaling software development requires cultural transformation. Next, we'll examine the value of Compliant Database DevOps (CDD) from three different lenses, then discuss the new ways that IT workers provide business value and drive organizational performance. We will then discuss global metrics to track your performance. Finally, we'll step through Redgate's paradigm for Compliant Database DevOps, which helps you identify where to focus at different points in your CDD journey in order to build value.

4. [What is DevOps](#), Donovan Brown, 2015



## Why can't we scale value by adding more engineers when a project gets behind?

You might wonder why DevOps is needed to increase business value. Why isn't the answer to delivering value faster simply a problem of hiring more software engineers?

Answers to this question go back to 1975, when Fred Brooks observed that "adding human resources to a late software project makes it later."<sup>5</sup> Estimating timeframes and schedules for software and IT projects is still incredibly tough. Once you get behind schedule, adding too many people to a project in an attempt to speed up work commonly has the undesired effect of slowing down the project. Fred Brooks observed that this occurs for a few reasons: software developers need ramp-up time to become productive, communication in larger groups has increasing overhead, and tasks may have a limited divisibility.

DevOps emphasizes practices learned from the Agile and Lean Development software movements which reduce these barriers to scaling software engineering teams<sup>6</sup>. Creating and disbanding teams as needed, and regularly moving members between engineering teams, reduces ramp-up time, by promoting constant habits to simplify, automate, and document procedures that get team members up and running. Communication overhead is reduced by giving team members visible displays which show work status, bottlenecks in the flow of work, and monitoring data that represents the current customer experience. DevOps also emphasizes decomposing tasks rigorously when planning all work and architecting loosely coupled systems. Having small, atomic tasks and decoupled systems already in place enables work to be more easily shared among a team which has suddenly increased in size.

DevOps is designed to help you implement the people, process, and tool changes that enable your technical and business teams to work together more effectively.

5. This is known as "Brooks' Law." See **The Mythical Man-Month: Essays on Software Engineering** by Fred Brooks, 1975

6. For a fascinating discussion of Brooks' law and DevOps, see **Accelerate: The Science of Lean Software and DevOps** by Gene Kim, Jez Humble, and Nicole Forsgren

# Lenses for examining the business value of DevOps

There's no single view of the business value of DevOps, because value varies based on perspective. Because of this, David Linwood discussed ROI via three "lenses" in *The ROI of Compliant Database DevOps*<sup>7</sup>: the lens of the CEO, the lens of the CIO/CTO, and the lens of the Manager.

Mirroring Linwood's approach, from a broad perspective, what is the business value of DevOps as seen through each of these three lenses?

## CEO

CEOs set corporate vision and are concerned with maximizing the value of the entire organization. CEOs want to have a faster time-to-market than competitors. They wish to have high customer satisfaction while avoiding unexpected costs from major failures. From the CEO perspective, the value of DevOps lies in its mission to continuously improve the delivery of value from software projects. That's because research shows that building valuable technology core to business operations helps companies gain more market share than they can by mergers and acquisitions<sup>8</sup>.

## CIO and CTO

CIOs and CTOs are shifting their focus: in the older view of IT organizations, the role of the CIO was to fulfill the requirements set by other business leaders with a reasonable quality of service and with an eye to cost savings. This role is shifting. The focus is now for everyone across all the silos of the business to provide and maximize value for end users - the organization's external customers. From the lens of the transforming CIO, the value of DevOps is its ability to iteratively improve the productivity and throughput of software developers and IT engineers in collaboration with others across many business silos. DevOps provides practices and global metrics that can be used to track and improve performance for everyone across these business units<sup>9</sup>.

## Manager

From the Manager lens, team performance is paramount. The value of DevOps lies in its ability to improve productivity while simultaneously decreasing burnout<sup>10</sup>, and in its ability to minimize the impact of failure on the organization's products and services, and on the team members themselves who work in the organization.

To unlock the value of DevOps, all of these leaders need to act as catalysts and supporters. This enables IT engineers and software developers to implement new ways of working. In collaboration with others across the business, they can deliver value to customers in a unified pipeline at an increasing rate.

7. [The ROI of Compliant Database DevOps](#), Redgate, 2017

8. James Bessen, 2017

9. [Accelerate: The Science of Lean Software and DevOps](#) by Gene Kim, Jez Humble, and Nicole Forsgren

10. Forsgren, Humble, Smith, and Frazelle, 2019

## The new ways in which IT specialists and software engineers create business value

As IT leaders and IT departments change their focus, the roles of specialist engineers such as database administrators, information security specialists, sysadmins, storage specialists, and network administrators are also changing.

Traditionally, in the older world of IT, these specialists have acted as gatekeepers for the production environment. Specialist engineers in traditional roles are the primary people responsible for the health of production environments. Their mission is mainly to respond to incidents and keep live systems running, so they usually encounter changes to production environments shortly before deployment. When the quality of production deployments suffers, the traditional recourse for these specialists is to attempt to slow down the rate of deployments by adding new layers of process onto changes just prior to deployment. As reactionary firefighters, there is little time for these specialists to improve the early stages of the software development lifecycle.

Increasing the ability of the organization to deliver value requires shifting the roles of specialist engineers. Specialists must shift from being the gatekeepers of production to consultants whose insights shape the software development process itself. Specialist production engineers hold valuable information learned from years of experience. They understand which systems are the most fragile, which type of deployments are the most prone to failure, and other patterns indicating when a change requires additional thought and a careful approach.

It's not enough to share this knowledge once, or to document the information. Instead, IT departments need to integrate both the knowledge and the operational expertise of specialist engineers into the early stages of software development. Doing so includes automating processes to validate code and test for risk factors before code is accepted into critical branches. It also involves automatically identifying when specialists should be included as code reviewers for critical or risky changes.

This can be a difficult shift for engineers to make. These workers have been trained for many years to be the guardians of production. Changing this role raises big concerns for them: if I'm no longer the primary person who is held accountable for service quality in my area, who is accountable? What protects the stability of the production environment?

The answer is global metrics for organizational performance.

## Global metrics for DevOps performance

Development and operations teams, along with business users across the organization, should all be globally accountable for both the tempo of deployments and production stability.

By giving all team members shared goals to increase the tempo at which value is delivered to the user, while maintaining quality services, DevOps removes the traditional conflict where development wants to speed up (to meet business demands for software delivery) while operations wants to slow down (to meet business demands for production stability). Instead, both teams focus together on improving their tempo, detecting and repairing problems before they impact end users, and increasing the resilience of production systems.

The Accelerate State of DevOps Report 2019 shares a set of four recommended metrics which indicate a team's DevOps performance<sup>11</sup>, in addition to availability metrics:

Tempo		Stability	
Deployment frequency	Lead time	Time to restore service	Change failure rate

By orienting all teams to track these metrics and work iteratively towards improving them, IT leaders align everyone toward shared outcomes that increase business value.

Cyril Lamblard, the Global Head of eCommerce at Nestle Nespresso, advocates that global measurements should be used even more broadly across business units, and that customer-centric global outcomes focus and unify the action of team members who might otherwise have disparate goals which cause friction<sup>12</sup>.

Once you have established these goals, the next step is to identify where you should begin. For code that manages changes to databases, "Teams that do well at continuous delivery store database changes as scripts in version control and manage these changes in the same way as production application changes."<sup>13</sup>

Redgate provides a framework for DevOps in the context of the database specifically, to get you started with continuous delivery. We will dive into this next.

11. [Accelerate State of DevOps Report 2019](#), Forsgren, Humble, Smith, and Frazelle, 2019

12. [Nespresso's Digital Transformation: From Vision To Operational Reality \(Live\)](#), What It Means: a Forrester Podcast, Forrester Research, Inc., 2018

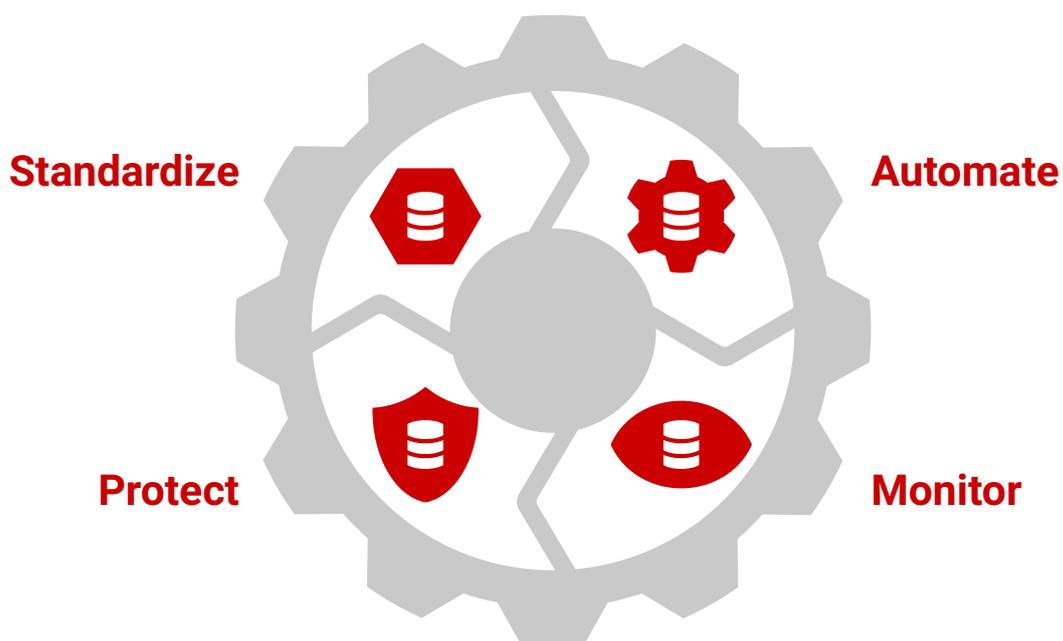
13. [Accelerate State of DevOps Report 2018](#), Forsgren, Humble, Smith, and Frazelle, 2018

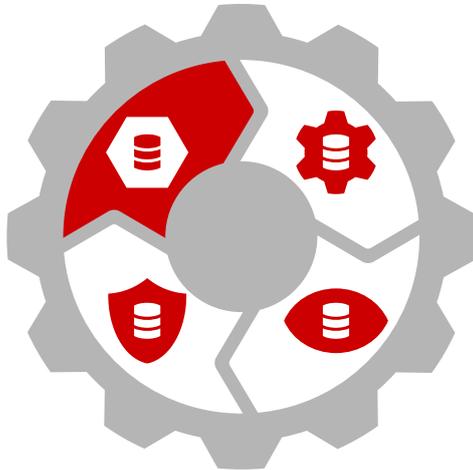


## Redgate's framework to iteratively increase value with Compliant Database DevOps

Redgate provides a paradigm that helps you establish the foundations of Compliant Database DevOps, then iteratively improve your performance as you continue your DevOps journey. This model includes four elements: Standardization, Automation, Monitoring, and Protection.

While establishing a version control system (VCS) is a fundamental step for team-based development that unlocks many capabilities in automation, each of these four areas contain valuable foundations for DevOps, and you may work through the four areas iteratively in any order.





## Standardize

Standardization brings invaluable practices to the database, practices which have improved quality and assisted in deployment tempo for application components for years.

An important step forward in standardization is to implement a VCS for database schema source code and data modification scripts. This provides the foundations for collaborating on code for a team, as well as critical auditing and accountability for code changes over time. While use of a VCS has been the default behavior for application code for years, 45% of database environments lack version control<sup>14</sup>.

It is important to bring an additional practice of modern development to the database. Engineers need environments that support innovation and allow experimentation for database changes. As Martin Fowler and Pramod Sadalage explain in *Evolutionary Database Design*,

**“People learn by trying things out, so in programming terms developers experiment with how to implement a certain feature and may make a few attempts before picking one. It’s important to be able to experiment in (a) private workspace and push to a shared area when things are more stable.”<sup>15</sup>**

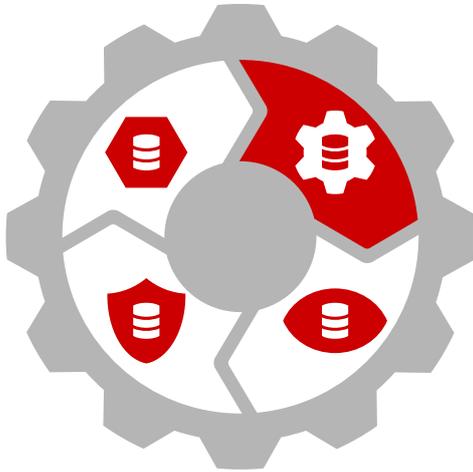
Combining version control with a private workspace, also known as a sandbox or dedicated development environment, empowers developers to experiment with coding patterns and learn safely, without impacting the work or productivity of others on the team.

Further standardization efforts include equipping development teams with a unified toolset to help with onboarding new team members, establishing coding standards and formatting guidelines, and streamlining the way the team locates, manages, and refreshes safe datasets for development work.

14. Redgate, 2019

15. [Evolutionary Database Design](#), Martin Fowler and Pramod Sadalage, 2016

## Automate



While automation is an important element of Compliant Database DevOps, sometimes people mistakenly believe that automated deployment is the goal of DevOps. That is not the case. The goal of DevOps is to enable the continuous delivery of value to end users, not to automate change delivery.

This is an important distinction, because mistakenly focusing too heavily on automating deployments to production environments may mislead you and may skew your understanding of your team's performance. Team performance should instead be measured by the global metrics we discussed above, which include markers for both tempo and stability.

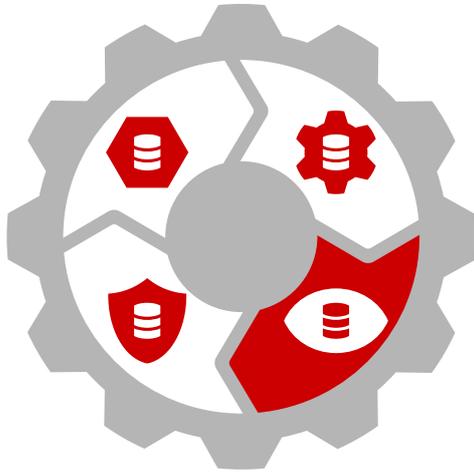
Automation is more powerful than something that only deploys software. It's also a powerful tool to assist in writing high quality code. Without the use of version control and automated code builds, it is common for database administrators to receive drops of code deployments which don't even parse properly – that is the comparatively impoverished state of traditional database development.

Automation in Compliant Database DevOps begins with providing critical code validation tools early in the software development cycle. Engineers are empowered to easily and routinely use tooling to validate that their code can be executed successfully.

Additionally, continuous integration and automated provisioning techniques for Compliant Database DevOps allow engineers to automate deployments against a production-like database in a safe environment. This can be built into the continuous integration process itself to ensure that realistic environments are automatically created for high-quality review of code changes.

The road to fully automating production deployments for database changes is not a short one. But understand that you can increase the business value for your organization dramatically by first focusing your automation goals on validating database code quality and provisioning development and test environments for database changes. While 92% of teams categorized as "Elite" performers by the Accelerate State of DevOps 2019 report automate their builds, only 69% have fully automated their production deployments<sup>16</sup>.

## Monitor



As organizations shift to deliver value to customers more rapidly than ever before, customers expect service providers to provide consistent, high quality experiences. A 2018 survey by Salesforce found that 63% of customers “expect companies to provide new products/services more frequently than ever before.” Additionally, 80% of customers said “the experience a company provides is as important as its products and services.”<sup>17</sup>

To meet these challenges, Compliant Database DevOps teams must have strong visibility into their customer’s experience. Monitoring fulfills multiple important functions to provide this visibility.

First, monitoring should recognize database deployments as an event and contextualize the point at which a deployment occurred in performance and utilization graphs and metrics. Displaying these metrics to all team members and referencing these metrics regularly ensures that deployments do not negatively impact the customer experience. Database monitoring also predicts and warns about factors that may cause customer-impacting incidents early, to either prevent outages or minimize the time needed to diagnose the issue and recover from it.

While lean principles for software engineering stress that monitoring data should be used to make business decisions on a daily basis<sup>18</sup>, this is even more important for Compliant Database DevOps because of the critical role that data commonly plays in an organization. If your data is not available, your customer’s entire experience is often dramatically impacted. If your data is corrupted, damaged, or lost, permanent damage may be done to the company’s brand, and in some cases the survival of the company is put at risk. These factors mean that careful monitoring of the production database must additionally validate the health of backups and the systems supporting high availability and disaster recovery.

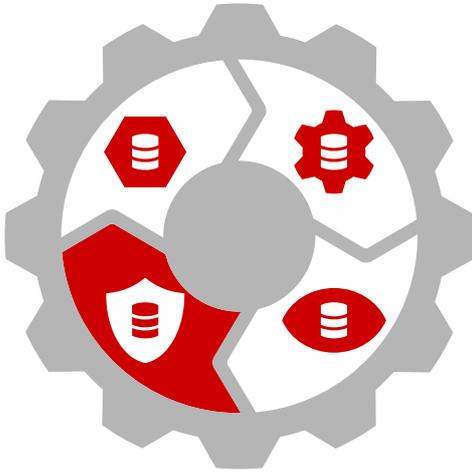
16. Forsgren, Humble, Smith, and Frazelle, 2019

17. [Salesforce.com](https://www.salesforce.com), inc., 2019

18. *Accelerate: The Science of Lean Software and DevOps*, Gene Kim, Jez Humble, and Nicole Forsgren, 2018

19. [The Developer Coefficient](https://stripe.com/blog/developer-coefficient), Stripe, 2018

## Protect



To enable Compliant Database DevOps, we must protect sensitive data by default in the software development lifecycle.

In 2018, Stripe surveyed more than a thousand C-level executives. When asked how much of a priority it is for upper management to increase the productivity of developers, 96% responded that it is a high or medium priority. Additionally, the report finds that “bad code” costs companies \$85 billion annually<sup>19</sup>.

Compliant Database DevOps focuses on improving both the quality of code and tempo of deployments in order to deliver more value to end users. As we have discussed, an important tool in ensuring quality database code is providing engineers with the ability to use production-like datasets early in the software development lifecycle to validate changes.

Production-like datasets increase productivity for software development and reduce the production of “bad code” in two important ways. First, realistic datasets enable a developer to validate that a query produces the desired results. Second, they allow developers to estimate the performance impact of changes to the schema of a database. Validating both the functionality and performance of code early on reduces the need to revisit and tune code later.

However, with the increase in regulations controlling how Personally Identifying Information (PII) is handled, as well as the increasing risk of data breaches, organizations cannot simply use copies of production databases everywhere for development. PII and other sensitive data must be de-identified (or “masked”) for use in non-production environments before datasets are made available to development teams.

As you build your capabilities for Compliant Database DevOps, your team must develop “guardrail” processes for provisioning datasets in the software development process. Guardrails allow your team to develop, validate, and deploy database changes at a regular tempo while only using safe datasets that minimize risk to the organization if a developer’s credentials are obtained by a malicious actor. These guardrails may be implemented by data masking and data virtualization technologies which support the rapid cloning of even large datasets.

Further protection efforts include enabling self-service of datasets, streamlining the refresh of available datasets to occur without manual intervention, and building up an inventory of the data estate through data classification. This inventory defines which production data needs to be de-identified before it can be delivered to development teams.

## Summary

Compliant Database DevOps is essential for IT leaders who must increase the amount of value which their business can deliver. To do this, IT leaders and departments are shifting from their traditional role as cost centers who fulfill requests, to technologists who collaborate across multiple business silos to innovate and consistently deliver value to end-users.

Global performance metrics enable IT leaders and business owners to drive performance across all business units, and to reduce friction between silos which traditionally have had conflicting goals. Teams can baseline these metrics today and immediately begin their Compliant Database DevOps journey.

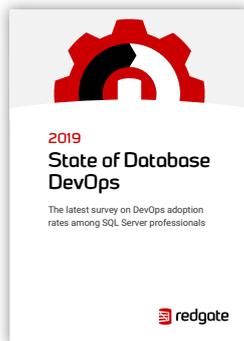
Redgate's Compliant Database DevOps framework consists of the elements of standardize, automate, monitor, and protect. This framework helps you establish the foundations of Compliant Database DevOps by taking a first pass through all four elements. Following this, iteratively move through all four elements repeatedly to identify the biggest current constraint which you should work to improve.



## Learn more

### Get started:

To find out more about the benefits of Compliant Database DevOps or learn how you can get your team started, visit [redgate.com/solutions](https://redgate.com/solutions) or contact us at [redgate.com/contact](https://redgate.com/contact)



### Dig into research:

#### [The 2019 State of Database DevOps Report](#)

Learn the main drivers for automating the delivery of database changes, the impact Database DevOps has on regulatory and compliance requirements, and how organizations are increasingly adopting Database DevOps alongside broader DevOps initiatives – and the advantages it brings.



### Read the whitepaper:

#### [Database DevOps: 6 tips for achieving continuous delivery](#)

Continuous delivery has changed the rules of the game. It's not just about moving from big bang releases to smaller, faster, more frequent releases. It's about putting in place DevOps processes so that reliable software can be deployed without problems, at any time.

## Acknowledgements

The ideas in this paper owe a great deal to the excellent body of research published by the DevOps Research and Assessment Group, led by Dr. Nicole Forsgren, Jez Humble, and Gene Kim. Their six years of research on IT performance, including the Accelerate State of DevOps Reports from 2015 to 2019 and their book Accelerate: The Science of Lean Software and DevOps provided a rich source of information as well as references to other literature cited in this paper.

## Bibliography

- Bessen, J.** (2017, December 1). Information Technology and Industry Concentration. Retrieved September 4, 2019, from [Boston University School of Law](#)
- Brooks, F.** (1975). The Mythical Man-Month: Essays on Software Engineering. Addison-Wesley.
- Brown, D.** (2015, September 1). [What is DevOps?](#) Retrieved September 4, 2019
- Forrester Research, Inc.** (2018, April 5). [What it Means podcast](#). Retrieved September 4, 2019
- Forsgren, N., Humble, J., & Kim, G.** (2018). [Accelerate State of DevOps Report 2018](#). Retrieved September 2019, from DevOps Research & Assessment
- Forsgren, N., Humble, J., Smith, D., & Frazelle, J.** (2019, August 22). [Accelerate State of DevOps Report 2019](#). Retrieved September 4, 2019
- Forsgren, N., Kim, G., & Humble, J.** (2018). Accelerate: The Science of Lean Software and DevOps. Portland, OR: ITRevolution.
- Fowler, M., & Sadalage, P.** (2016, May). [Evolutionary Database Design](#). Retrieved September 2019
- Gartner, Inc.** (2017, April 27). [2017 CEO Survey](#). Retrieved September 4, 2019, from Gartner
- Gartner, Inc.** (2019, May 8). [Gartner Survey Reveals That CEO Priorities Are Slowly Shifting to Meet Rising Growth Challenges](#). Retrieved September 4, 2019, from Gartner
- Redgate.** (2017, November 22). [The ROI of Database DevOps](#). Retrieved September 4, 2019
- Redgate.** (2019). [State of Database DevOps Report 2019](#). Retrieved September 4, 2019
- Salesforce.com, inc.** (2019, June). [State of the Connected Customer Report 2019](#). Retrieved September 2019
- Stripe.** (2018, September). [The Developer Coefficient](#). Retrieved September 2019

