

# Filling the Data Gap in DevOps:

How Datical and Delphix Accelerate the Journey  
to Digital Transformation



# Contents

- Executive Summary . . . . . 2
- Introduction . . . . . 3
- The Challenge . . . . . 4
- The Solution . . . . . 5
- Technology Overview: Delphix . . . . . 6
- Technology Overview: Datical . . . . . 7
- The Benefits of a Combined Solution:  
Delphix and Datical. . . . . 8
- Case Study: Chocolate  
and Peanut Butter . . . . . 9
- Summary . . . . . 9

# Executive Summary

While many IT teams have been successful in automating a number of manual processes in modern software development, the same has not been true for the data underpinning the system. Today, provisioning, versioning, and aligning data to database code is still a largely manual process that adds considerable friction to application projects and slows the pace of innovation.

This paper highlights the challenges companies face when it comes to managing data for application projects and how companies can accelerate their goals for digital transformation by refocusing attention on the data gap.



# Introduction

Software is reshaping industries, from healthcare to finance, to retail and beyond. The pace of change is accelerating, and companies must quickly deliver applications in order to survive in the digital era. Speed is the new standard in the software economy.

To achieve the goal of faster time-to-market, companies continue to adopt faster more iterative development methodologies. These organizations have replaced age-old waterfall development methodologies with agile practices. From an infrastructure perspective, they've also invested in modern architectures and cloud technologies to achieve higher efficiencies. Some companies have combined these two approaches by adopting DevOps—investing in new tools and processes such as infrastructure automation and continuous integration—to clock even faster speeds.

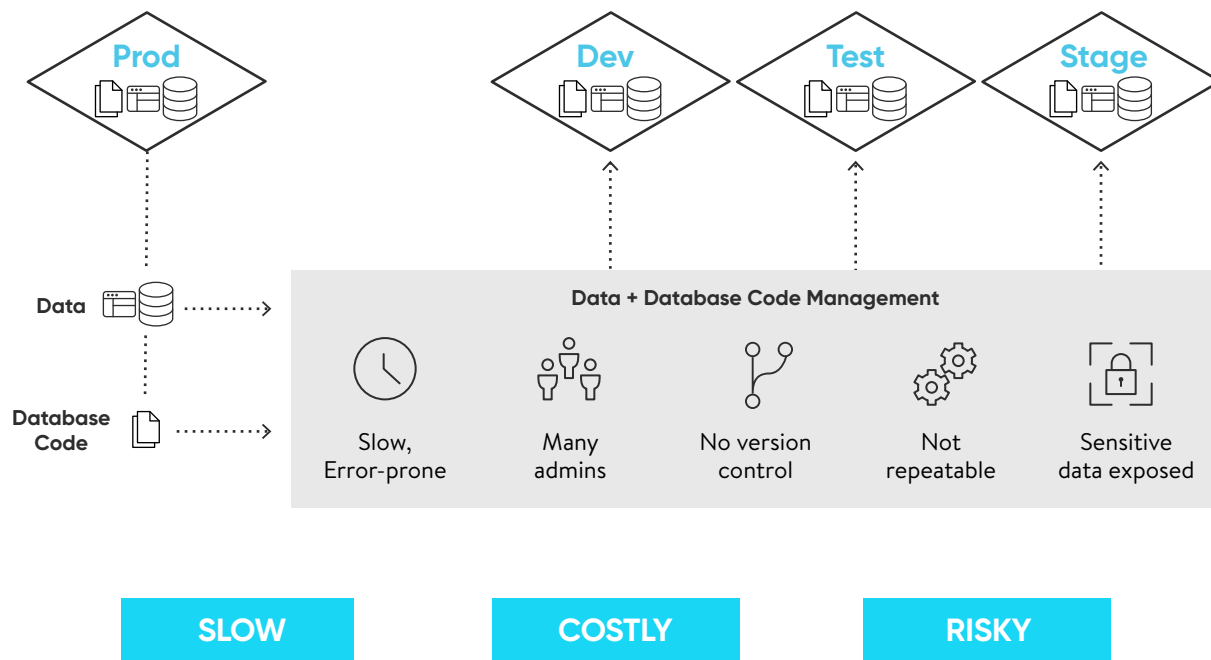
But as companies adopt faster development methodologies, a new constraint has emerged in the journey to digital transformation. Data has long been the neglected discipline—the weakest link in the tool chain—with provisioning times still counted in days, weeks, or even months. Most companies, even those using the latest DevOps automation tools, still manage and deploy database changes manually, further anchoring development teams. Put differently, most organizations attempt to support modern, agile development environments with half-century-old database and data management processes and procedures—a result akin to mounting a modern Ferrari on spoked Model T tires. It's a great way to get nowhere fast.



# The Challenge

Today's development teams are increasingly hamstrung by the speed and quality of data. Developers require fast, high-fidelity data, but their requests often go unmet because environments are expensive and time-consuming to create. As a result, they're forced to work with low-quality, stale, or incomplete data—leading to adverse consequences such as more time spent analyzing and resolving data-related defects instead of coding. Behind

the scenes, IT operations teams are constrained by the slow, inefficient process of extracting, copying, and moving data from system to system. For systems containing sensitive data that must also be secured, environments can be even more complex and costly to provision. In some cases, constraints in the availability of environments and data can lead to delayed releases or production downtime, which can have a material impact on the business.



Syncing both data and database schema changes with application releases has been a challenge.

To make an already antiquated process worse, each new version of an application requires structure or logic updates to the database—including adding or changing tables, columns, or stored procedures. This means application developers need a mechanism to update the database schema to align with their version of the application. After provisioning a new environment with production data for an existing development branch, for instance, database teams might need to apply committed database structural changes to the data and also potentially inject any required synthetic data. Deploying database code, however, is often slow and error-prone. Many IT teams struggle to maintain and align database code with the application code because their schema management processes are not repeatable. Even if a change is virtually identical to a thousand previous changes, it must be approached as a new process, which can lead to inconsistent configurations. A single failure can lead to even more failures downstream, resulting in expenditures on the order of hundreds of hours spent troubleshooting. All in, database schema management further hinders development teams from working at an agile pace.

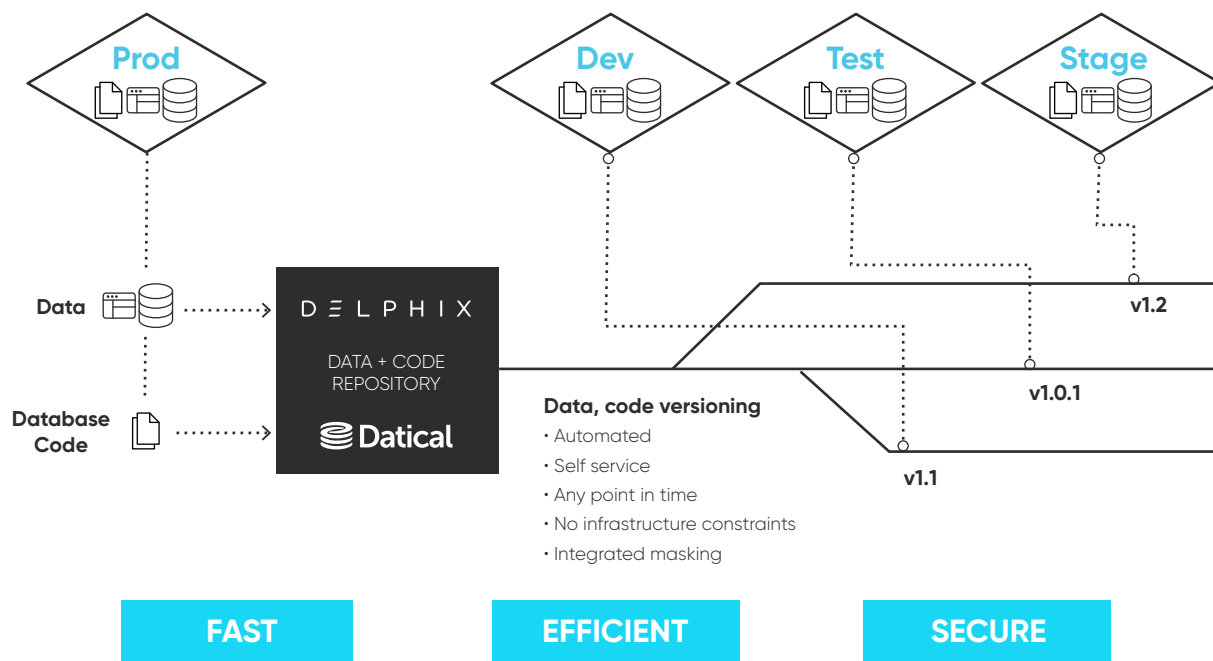
Keeping both data and database schema changes up-to-date with application releases has always been a challenging task. In an environment where speed counts everyday, it's no longer adequate to add more people and infrastructure to solve these challenges. There's an increasingly pressing need for a mechanism that delivers high-fidelity, usable data in a fast and repeatable manner.



# The Solution

The solution is to focus on data. The ability to quickly provision, refresh, and tear down environments—automating the management of both data and database code—is a fast, simple, and non-disruptive way to pave the path towards DevOps. By taking a data-first strategy, companies can break the logjam in development, enabling teams to achieve faster time-to-market.

In an automated system, data environments are provisioned as needed in a matter of minutes. Developers can access the latest production data without the wait. They can also recreate previous data sets in minutes, essentially creating an instantly accessible history of their data without versioning complexity.



Taking an innovative approach to data and database code versioning with Delphix and Datical.

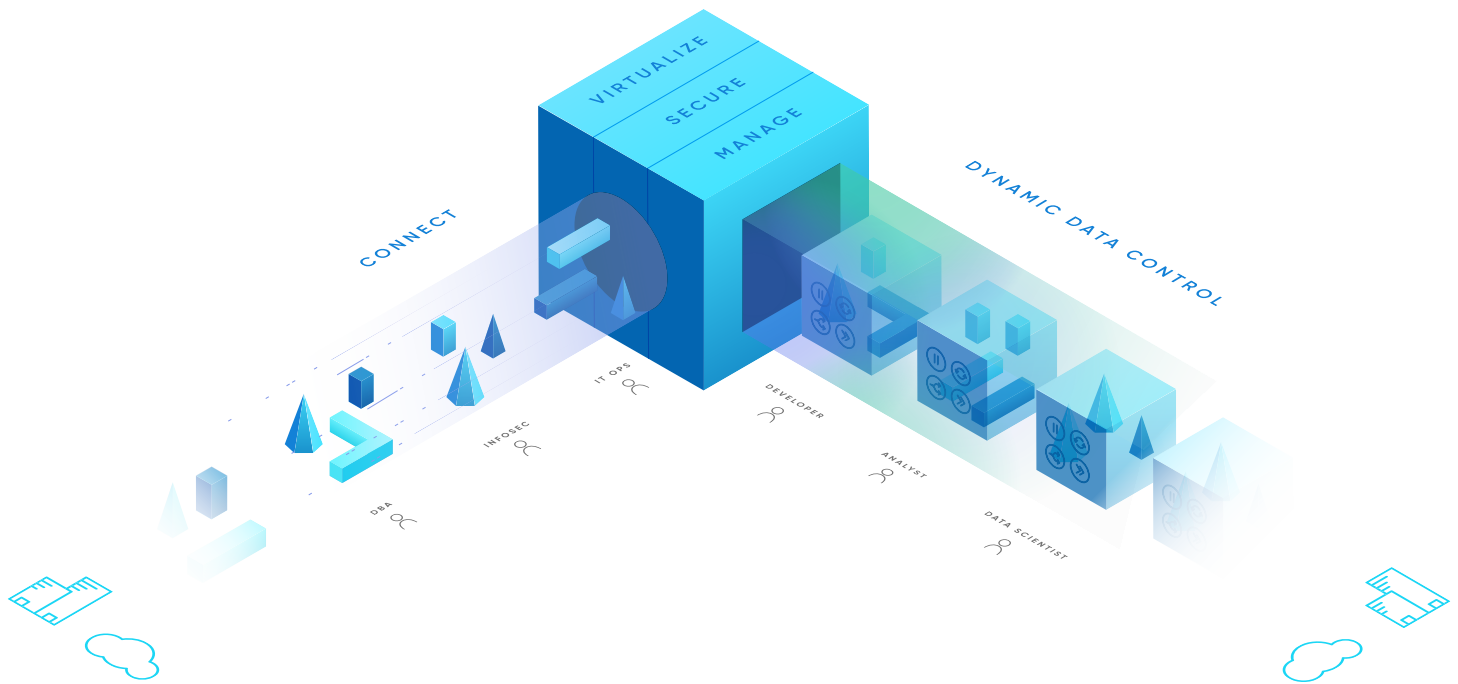
In parallel, database releases are managed by development teams in the same way that application code is managed, transforming schema management into a consistent and repeatable process. Database releases are more routine and less risky; IT teams can forecast impact prior to deploying changes, eliminating the possibility of disruption. The ability to track every change and create a history of database code changes in an automated fashion means IT teams can also comply with audits in a simple and risk-averse manner.

Delphix and Datical offer a combined solution to accelerate data delivery and automate database releases, increasing the speed and quality of the total build environment process. By further providing self-service controls through a continuous integration layer such as Jenkins, Delphix and Datical enable developers to version both their data and their database schema changes—alongside application changes—like code, which means more autonomy, less dependence on infrastructure teams, and faster releases.



# Technology Overview: Delphix

The Delphix Dynamic Data Platform allows companies to instantly move and secure data with integrated masking, enable self-service access, and automate manual processes. It runs on any server, storage, and supported hypervisor either on premises or in a cloud environment, and operates by performing five key steps:



## Connect:

Non-disruptively collect data from databases, applications, and file systems. After compressing this data, the platform stays synchronized with sources by recording all changes over time.

## Virtualize:

Through intelligent data block sharing, create virtual copies of the sources that are space-efficient, portable, and fully readable/writeable.

## Secure:

Automatically apply data masking to scramble confidential information in copies, and define policies that determine retention characteristics and user privileges.

## Manage:

Quickly provision secure data copies to target environments with functionality to audit, monitor, and report against access and usage.

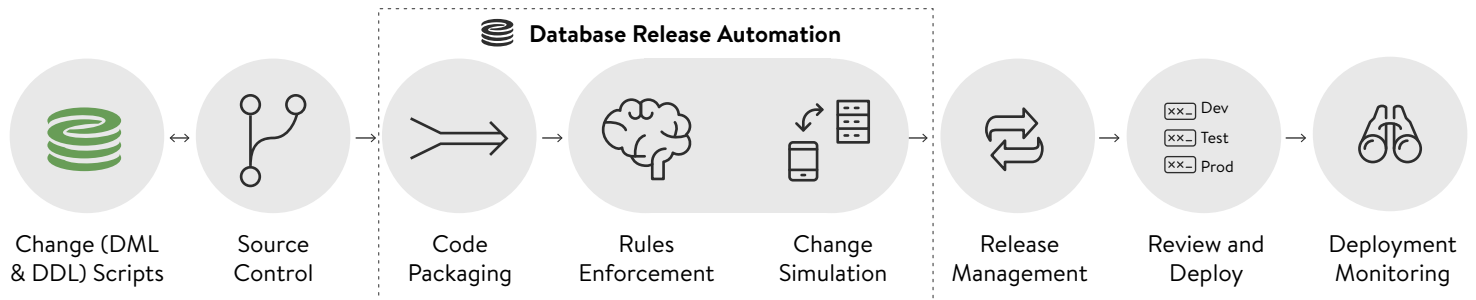
## Self Service:

Provide developers, testers, analysts, or data scientists with controls to manipulate data at will: refresh, rewind, bookmark, branch or share data.



# Technology Overview: Datical

Datical is a database release automation solution that enables companies to improve and simplify the application release process by allowing them to treat the database code that manages structure and logic changes just like application code. It supports databases that run on premise or in the cloud and operates based on five key capabilities.



## Database Change Simulation

Datical employs a virtual model of the ideal state of a database in order to track and affect change. Datical simulates proposed database changes against the current state of an environment prior to making any changes. When it's time to deploy, Datical reconciles the target environment with the ideal state represented by the virtual model, ensuring a complete and correct ideal end state regardless of the version from which you start.

## Rules Enforcement

Datical automatically enforces DBA rules across all proposed database changes. This ensures proposed database code changes are safe and in compliance with an organization's database standards before they are deployed. In addition, automated rules enforcement enables application developers with a self-service validation capability that provides instant feedback on proposed changes. If the change passes the rigorous checking performed by Datical, it is propagated immediately.

## Code Packaging

When the process of introducing application changes is separated from the process of introducing database changes, it's easy for dependent changes to become out of sync. With Datical, application teams can check their database change scripts into their application's source code repository. Datical

retrieves, validates, and labels database changes, making it easy to trace changes back to the corresponding application task or business requirement. The result is a complete package of application and database changes that can be promoted to or removed from an environment or planned release in lock step.

## Deployment Monitoring

Manually tracking database change script execution across all environments in the software development lifecycle is not an ideal approach. Add in database deployment automation and the task becomes difficult, if not impossible. Datical collects detailed information about every database deployment it performs and automatically logs it to a centralized database, eliminating the need for spreadsheets and human error inherent in manual entry. In turn, database audit becomes quick and easy.

## DevOps Integration

Many industry standard tools exist to implement Agile and DevOps practices. Datical includes out of the box integrations and interfaces enabling integration with any agile automation framework. Datical integrates with commercially popular DevOps tools such as Git, Subversion and Microsoft TFS for Source Control, Jenkins and Bamboo for Build Automation, and IBM UrbanCode, Xebia Labs XL Deploy, CA Release Automation and Microfocus Serena for Deployment Automation.



# The Benefits of a Combined Solution: Delphix and Datical

Delphix plus Datical unlocks the full potential of DevOps. The Delphix Dynamic Data Platform makes it easy to create instant masked and unmasked copies of data from a data time machine and Datical enables the seamless integration of schema changes across those data copies. The four key benefits of an integrated Delphix and Datical for DevOps and non-DevOps teams alike include:

- **Faster development.** Developers can easily access and control data in development and test environments using Delphix. Delphix can also deliver masked copies of those production datasets, by policy, ensuring sensitive information is not migrated into unsecured zones. Providing developers with a copy of actual production data results in more bug-free code. In conjunction, Datical ensures that the most recent database schema changes are deployed to all environments. Datical can identify the deltas between production databases and development, helping to spotlight potential problems early in the development process. This benefit eliminates the typical back-and-forth between DBAs, development, and QA engineers during the process of remediating errors. The net result is decreased development costs and faster time to market.
- **Higher test integrity.** Delphix enables developers to access the latest production data, leading to fewer missed test cases and defects that reach production. In conjunction, Datical makes it easy to update the database schemas being used by developers and ensure they match the application currently undergoing testing. For most IT shops, the ability to test an update mechanism provides an exponential boost of confidence that the application will work as designed once it's installed in the production environment. Datical also facilitates the automation of test setups and helps to enforce uniform standards, leading to a decrease in production issues.
- **Less contention and fewer conflicts.** A monolithic database that all developers must share is a decadesold paradigm that still exists at most organizations. It can lead to a common problem where each developer's work potentially interferes with the work of other developers. Using Delphix, each developer (or team) can control and version their own data. They can also test in isolation using their own copy of data, without impacting other developers or teams in a shared environment. Datical can then be used to assure that the database incorporates the latest schema changes. Multiple databases, created for multiple developers, can also be merged into a new database—a capability virtually impossible to accomplish with manually managed SQL scripts. The use of Delphix and Datical eliminates wait time and provides an efficient means of identifying and resolving conflicts early, which is particularly valuable for departments that are responsible for multiple applications or support multiple scrum teams with multiple release trains.
- **Easier production hotfixes.** Production issues requiring immediate attention are highly common. But it's not uncommon for teams to have difficulty reproducing a production issue in a development environment. Delphix eliminates that difficulty by providing a means of instantly re-creating production data. Datical solves the challenges that might persist later on, such as manual database changes that go undocumented in the race to restore full functionality to production as soon as possible. If database changes stand in isolation and have not been replicated in the development and test environments, Datical isolates the fix, creates a delta, and generates a forecast by detailing the anticipated impacts of the changes. Database schema changes can then be intelligently deployed to production. Schema changes will also be deployed to non-production environments, avoiding configuration drift. And Datical will ensure that the change is smoothly merged with future releases. If the next production push contains the hot-fix database changes, Datical will recognize that the changes are already in production.





# Case Study: Chocolate and Peanut Butter

Sometimes a providential pairing delivers outstanding benefits. That certainly happened when Harry Reese first tinkered with combining chocolate and peanut butter in the 1920s. And it happened when one of the world's top five financial services companies combined Datical with Delphix.

The DevOps team at this large enterprise company was faced with an ironclad rule: no production data younger than six months could be used for development or for testing. Protecting customers' sensitive financial data is obviously important, but the rule posed problems for DevOps. In one instance, a mutual fund had closed within that six-month window, creating a bit of chaos for the DevOps team that was left in the dark about that significant change.

That sort of problem won't happen again. Delphix now enables fast provisioning of current production data, while replacing sensitive information with realistic, fictitious data. And Datical helps to keep the database schema in synchronization. In its integrated test environments, this company now runs tests using high fidelity, masked data and data structures that virtually mirror its planned production deployment. Even when a production push involves more than 100 steps when performed manually, using Datical to automate database deployments means that the deployment to production will occur exactly as it did in test. The net result is that this company has nearly eliminated database-related issues in their production deployments.

**Just like chocolate and peanut butter, Datical and Delphix stand alone just fine. But the combo of the two is more powerful—as many DevOps teams would testify.**

## Summary

There are many reasons to address the automation of data and database schema management. For organizations taking a DevOps approach and considering continuous integration and delivery, it's crucial to fill the data gap. Even for companies new to DevOps, it can be advantageous to start with data. Changing the way developers code, manage builds, perform migrations and deploy code can create great value, but not without the initial hurdle of fundamentally changing an existing process. Solving for data and database schema management with Delphix and Datical is the opposite. By partnering with the teams, IT teams can transform a highly manual process and unlock immediate value without disruption, paving the way for DevOps and digital transformation.

For more information visit [www.delphix.com](http://www.delphix.com) and [www.datical.com](http://www.datical.com)

The Delphix logo and design are registered trademarks of Delphix Corp. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

The Datical logo and design are registered trademarks of Datical Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.