DATAOPS VS DEVOPS: WHAT'S THE DIFFERENCE?

In today's technology landscape, DevOps continues to be a popular methodology that needs no introduction. Given the success of DevOps, emerging methodologies have embraced DevOps features to extend into niche areas of software development.

DataOps is one model that was created to help data management teams harness the power of automated data orchestration to develop intelligent, data-driven systems. The DataOps discipline introduces agile principles into data analytics, streamlining how data-centric applications are designed, developed, and maintained.

This article delves into the fundamental differences between DevOps and DataOps methodologies, similarities, and use-case benefits for each.

DataOps vs DevOps

While some presume DataOps as *DevOps for Data Science*, the two methodologies have differences in how they implement various stages of the development lifecycle. Though both emphasize agility and collaboration, they focus on different areas of business, and as a result utilize different approaches and pipelines.

So, let's explore each methodology to uncover their fundamental differences.

What is DevOps?

A DevOps model implements a set of practices and tools that enable the collaboration of development and IT operations teams throughout the entire software development lifecycle (SDLC).

While doing so, DevOps extends agile methodologies by enforcing automation beyond the build phase to integrate operations as part of a seamless workflow. Automation enforced by DevOps essentially encompasses the build-test-release cycle with Continuous Integration and Development (CI/CD) pipelines, enabling quicker time-to-market without impacting code quality.

The methodology promotes randomly sequenced iteration throughout the software's lifecycle, enabling rapid application development and delivery.

(Understand the importance of DevOps automation.)

Why people adopt DevOps

DevOps tools and processes enable quicker innovation, faster time to market, and increased bottom-line. With DevOps, organizations typically experience a number of benefits, including:

- **Improved communication and collaboration.** DevOps breaks operational silos between application development, release management, and operations teams, allowing for easier collaboration and sharing of resources.
- **Cost savings.** DevOps shortens delivery cycles, which reduces the cost of maintenance and upgrades by leveraging economies of scale through efficient CI/CD pipelines.
- **Reduced disaster recovery times.** DevOps implements the build-test-deploy cycles in small, managed batches that help minimize the impact of deployment failures, bottlenecks, and rollbacks.

Elements of DevOps

Initially built on a collection of ideas of how to improve application development workflows, DevOps has now grown into a full-fledged ecosystem with defined standards and tools. Some critical elements of a DevOps framework include:

- **Automation.** DevOps includes tools to perform repetitive and manual tasks automatically that require minimal human intervention.
- **Infrastructure-as-Code (IaC).** An essential aspect of implementing DevOps is to rely on better utilization of infrastructure resources. To do so, DevOps teams manage IT infrastructure using a descriptive model of code binaries that enable deploying and testing infra components in a simulated production instance.
- **CI/CD.** DevOps follows an iterative model that intertwines continuous development, testing, and deployment to enable automation, security, and agile development.

A typical DevOps workflow

DevOps is a cyclical process that constantly iterates between five stages:

1. Business vision blueprint
2. Build operations (Code development and testing)

3. Integration
4. Deploy (Release of the application)
5. Operate (Ongoing run and maintain)

The operate phase also loops usability feedback into the workflow that helps refine the business vision for responsive and iterative application development.

What is DataOps?

Modern applications and business systems continuously generate large amounts of data. DataOps is a methodology that outlines focused practices, tools, and frameworks for enterprise data management. The methodology standardizes the technological and cultural changes that help organizations to:

- Reduce the cost of managing data
- Improve data quality
- Enable faster time-to-market for data-centric applications

While doing so, DataOps bridges the gap between data collection, analysis, and data-driven decision-making, allowing organizations to efficiently deliver analytical insights for improved business value.

(Learn about BMC's approach to DataOps.)

Hear BMC CTO Ram Chakravarti share why traditional data and analytics approaches have fallen short and how DataOps can rapidly turn new insights into fully operationalized production deliverables that unlock maximum business value.

Benefits of DataOps

DataOps aims to augment the quality of data analytics by reducing the duration of data lifecycle and improve the quality of data analytics. Some benefits of the DataOps methodology include:

- Automating manual data collection and analytics processes
- Continuous monitoring of the data pipeline
- Isolation of production data
- Centralization and sharing of data definitions
- Enhancing the reusability of the data stack
- Enabling controlled data access

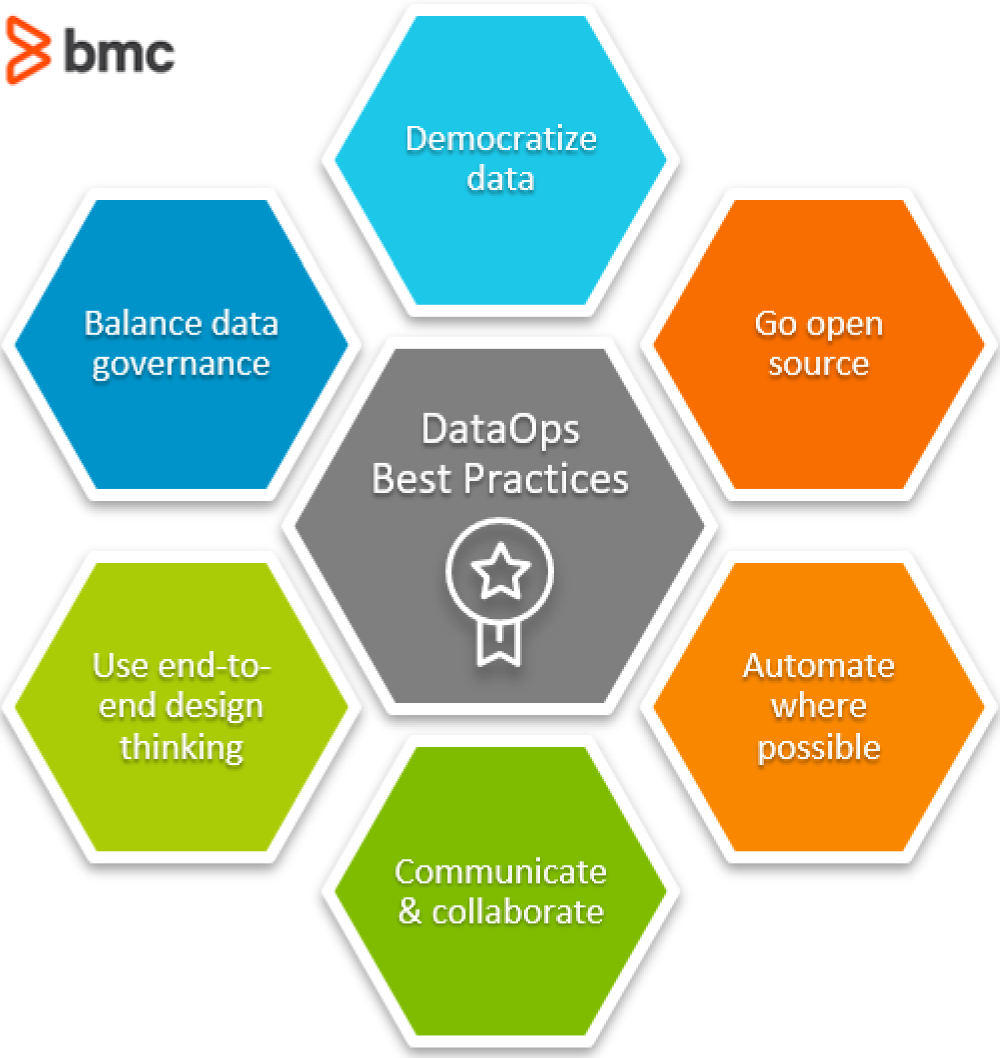
DataOps in modern app development

Organizations adopt artificial intelligence and machine learning models into their digital products and services for enhanced analytical insights and improved customer experience. The DataOps model specifically helps data scientists, ML engineers, and analysts to create models that support the end-to-end requirements of AI/ML frameworks.

Some critical areas of modern application development that benefit from DataOps include:

- Self-service interaction

- Data governance and curation services
- Log and event monitoring
- Vulnerability scanning
- Search and indexing
- Market analytics



DevOps vs DataOps

Both DataOps and DevOps' fundamental goal is to transform a product development lifecycle through enhanced agility and automation. Some similarities between the two methodologies include:

- Both employ agile methods to shorten delivery lifecycles
- Both enforce cross-functional collaboration between multiple teams
- Both utilize a multitude of automation tools for faster development

The main differences between the two methodologies include:

Quality factor

DevOps mainly focuses on developing quality software by shortening development cycles. DataOps emphasizes the extraction of high-quality data for quicker, trustworthy BI insights.

Delivery automation

DevOps focuses on automating version and server configurations.

DataOps focuses on the automation of data acquisition, modelling, integration, and curation for high-quality data delivery.

Collaboration

DevOps seamlessly integrates development and operations teams for rapid delivery while DataOps connects business leaders, IT development, and data analytics teams for quicker data processing.

Summarizing DevOps & DataOps

The current tech landscape is highly dynamic. To gain a competitive edge, businesses rely on applications that are highly scalable, efficient, and secure.

While efficient applications act as essential enablers for overall organizational efficiency, it is equally critical for organizations to adopt the right model that helps develop applications that are agile, efficient, and secure.

Among the list of various software development methodologies, DevOps continues to be the most popular choice. On the other hand, DataOps focuses on data-based application delivery.

Both DataOps and DevOps deliver advanced innovation and competitive advantage for firms looking to improve their development lifecycles. The two methodologies, however, differ in how they implement the build-test-deploy stages of software development. Keeping the basic difference aside though, both of their goals is to enable a highly efficient, automated framework that relies on team-level collaboration and comprehensive automation of development workflows without compromising application reliability.

Related reading

- BMC Machine Learning & Big Data Blog
- BMC DevOps Blog
- Top DevOps Trends Today
- Data Architecture Explained: Components, Standards & Changing Architectures
- Data Ethics for Companies
- Book Review of *The Phoenix Project*: DevOps For Everyone