

# Simplifying Cloud Database Access and Why It Is Important



## Introduction

In modern business, data is, as the saying goes, the new oil that drives success. As such, easy access to data is essential. The challenge is that due to the way modern applications are developed and deployed, providing access often becomes a complex and time-consuming process.

The scope of the problem is enormous and growing. The databases are being used for both transactional and analytical use cases that are at the heart of business operations in most companies. More organizations are shifting towards adopting databases on containers and Kubernetes. Many plan to deploy cloud-native databases or consume a Database-as-a-Service (DBaaS) offering from their cloud providers or favorite database vendors. An IDC survey<sup>1</sup> found that roughly 90 percent of enterprises are either actively moving production data to the cloud, adopting cloud databases for new workloads, or have plans to do so within the next three years.

Fast and easy access to these various cloud databases is critical for business success. If providing access is not done efficiently, everyone is unhappy. Database administrators must divert from their main chores to provide access credentials on a case-by-case basis. IT Ops teams must deal with managing control over multiple applications, clouds, and databases. And data scientists, data engineers, and developers become increasingly frustrated as their work is delayed while they are waiting for access.



## Limitations of traditional cloud database access approaches

While there are many challenges to implementing databases and DBaaS for modern, cloud-native applications, one that continually pops up is handling access credentials.

All groups involved, including database administrators, IT Ops, and application developers, have different issues when it comes to providing and managing access. For example, database admins need to gain control over which developers access various database services. Developers need to create unique access configurations for each application and database they work on. And IT Ops needs control and visibility across all database offerings.

Traditional approaches to providing access worked well when new applications and updates were made on an annual basis, development cycles were not so fast-paced, databases were on-premises and company-managed, and very few databases were involved. Fast-forward to today, and every one of those factors that made database access manageable is less likely to exist.

---

<sup>1</sup>IDC: Market Analysis Perspective: Worldwide Database Management Systems Software Market, 2021, September 2021. Doc US482 47521  
<https://www.idc.com/getdoc.jsp?containerId=US48247521>

But now, with so many more developers needing access to multiple cloud database resources, these methods are inefficient. There is no centralized visibility and control. They lead to sprawl and redundancies, making it harder to get a clear and reliable view of who's using what. Worse, traditional methods of providing access take more time than many developers are willing to wait. They resort to shortcuts, such as sharing credentials through Slack or email.

The result of such inefficiency is that companies lose the ability to monitor and manage application usage. They can lose control over access, and that can have serious compliance and security consequences. Additionally, if users freely share credentials, that can lead to unconstrained use of DBaaS services and redundant instances of those services.

## What's needed?

These factors are driving a need for a new database access model. A suitable solution should include benefits for both developers and operations.

**With regard to developers,** there are several characteristics that must be incorporated. A solution must be agile and offer a self-service model to provide efficient access to database cloud services.



Any solution for developers must support easy access methods. For example, services would be provided using a simple user interface and APIs for automation and, specifically, via automation within traditional application development environments such as those supporting CI/CD. A solution must also provide fast and simple connectivity, allowing developers to focus on their code.

Such capabilities help address database access challenges and lower the toil and frustration developers often face. In particular, a solution supporting those capabilities enables faster onboarding and higher productivity.

**With regard to operations,** again, there are numerous characteristics that would prove valuable. An appropriately selected solution would offer greater governance and control through a standardized approach to access across multiple DBaaS and database services vendors. It would also provide visibility over what's going on. The operations side of the house would know which services are being used and how they are being used.

Such a solution would allow database administrators and IT Ops to grant access control to developers and not have to answer many IT tickets. It would also allow a company to select database services more freely and then create access configurations that developers can leverage without having to know specific access control credentials.

## Teaming with a technology partner

Today, users are frequently mixing multiple methods of integrating database services into their applications, with a combination of cloud databases, databases deployed directly through Kubernetes, and connecting to virtual machines outside of the cluster running database services.

There are many ways a company might set out to implement a simpler database access solution to meet the needs of modern application requirements. But the last thing most companies need today is another do-it-yourself development effort.

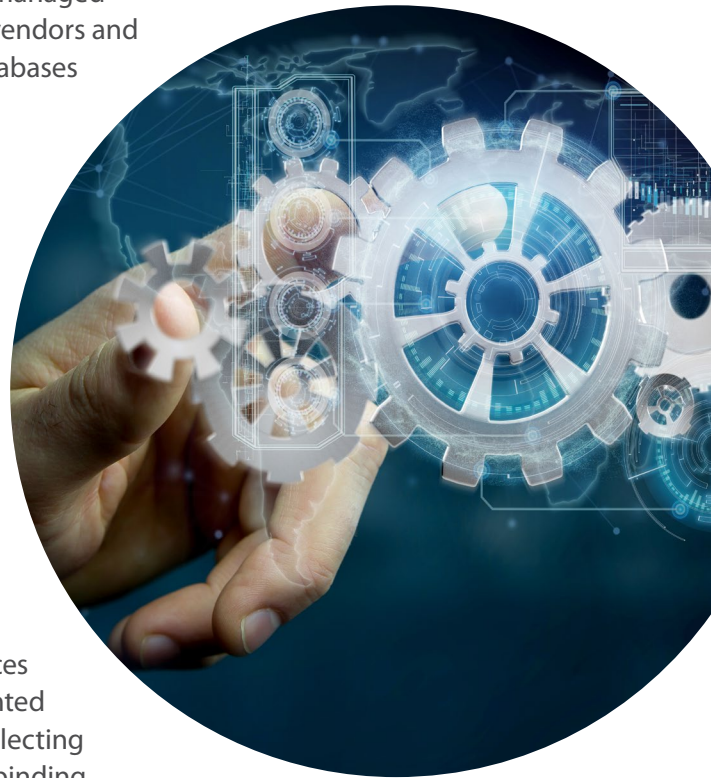
Increasingly, an alternative is to work with a technology partner with a proven solution and expertise in the market. Enter Red Hat and Red Hat OpenShift Database Access (RHODA), which is designed to simplify and standardize how teams provision, monitor, and connect to cloud-hosted partner database services across their environment.

Red Hat OpenShift already supports many databases, from self-managed cloud-native databases to DBaaS offerings from both database vendors and cloud providers. RHODA simplifies access to managed cloud databases directly from OpenShift cloud services to accelerate how quickly developers bring database-centric applications to production.

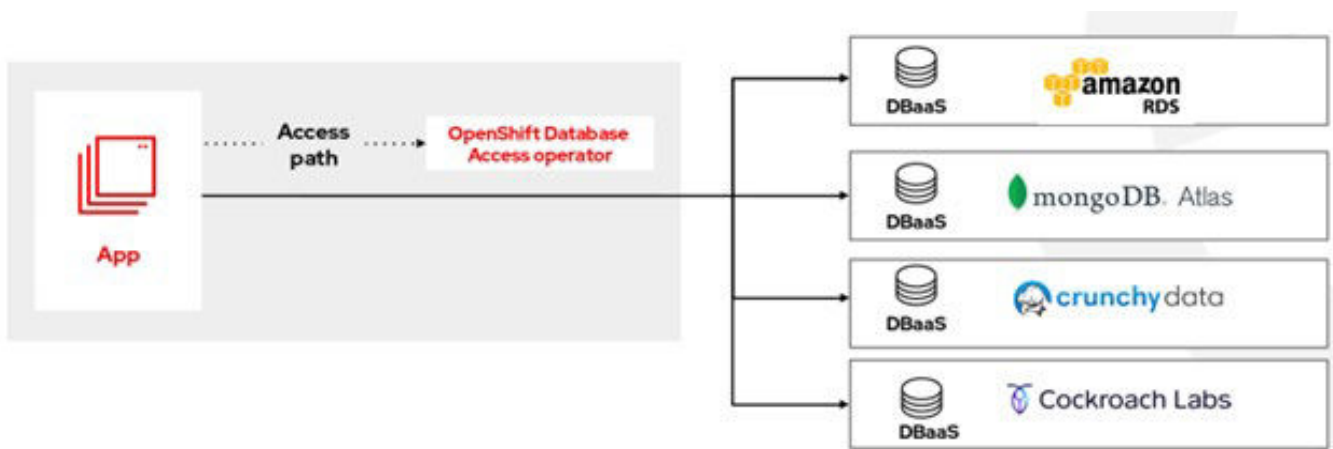
RHODA enables administrators of OpenShift cloud services to set up connections in managed OpenShift Kubernetes environments, enabling administrators to set up connections to DBaaS offerings from different providers. Developers get access via a self-service process. And administrators gain the centralized view and access control needed to manage many connections at scale.

RHODA leverages the Kubernetes construct of operators to bind applications to database instances. IT operations can launch a database discovery process based on resource permissions to identify the database instances that can be accessed. Once established, developers can see those new database services from the catalog on their OpenShift developer console, represented as a UI tile for each database registered with the system. After selecting a database service instance, the developer can drag and drop a binding connector on the OpenShift developer UI to establish database connectivity to a Kubernetes pod.

Red Hat OpenShift Database Access is a cloud service that is an extension to two OpenShift cloud services: OpenShift Dedicated and OpenShift Service on AWS. Initially, it supports a number of database cloud vendor offerings, including MongoDB Atlas, Crunchy Data's Crunchy Bridge, Amazon RDS, and CockroachDB Dedicated, a distributed SQL database service.







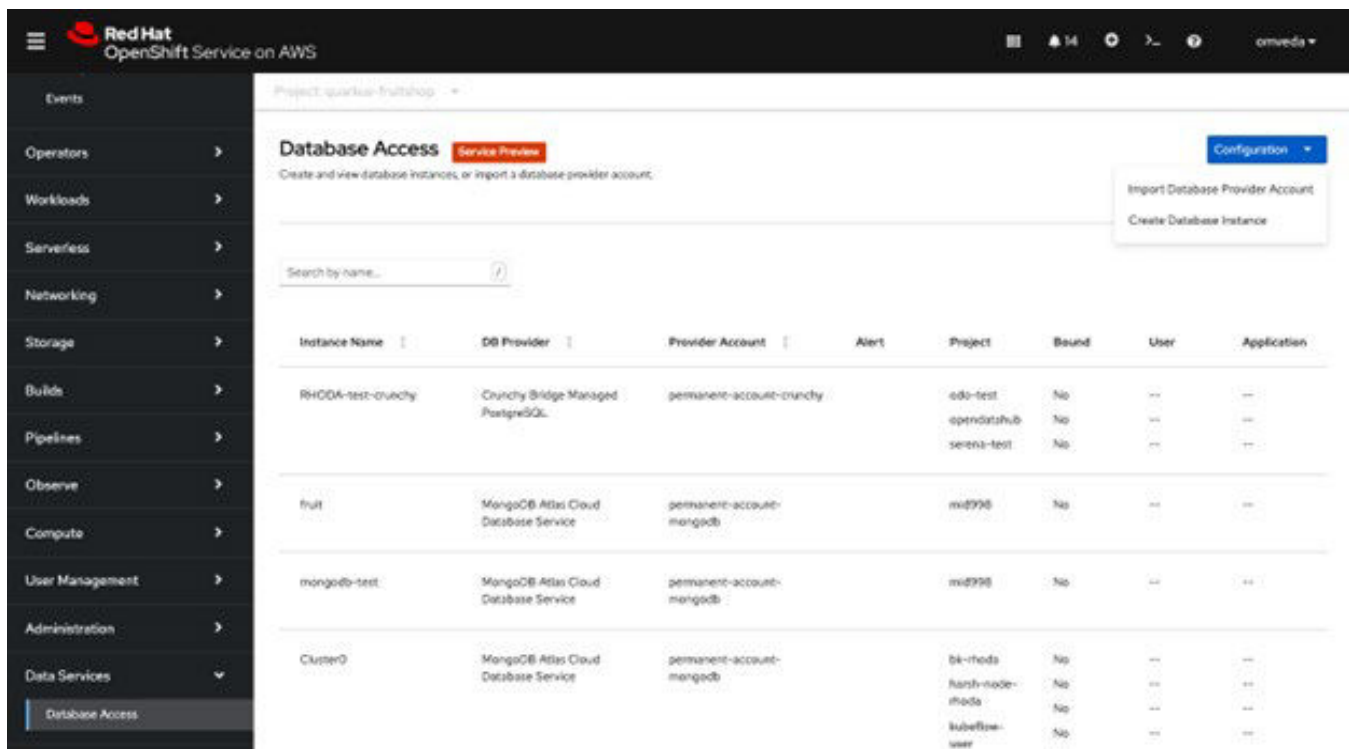
From a database admin perspective, RHODA helps to bring low-touch administrative experience to provision and manage access to external database services. From a developer perspective, RHODA helps to connect applications to database services with a consistent and predictable experience.

The solution offers an easy way for developers to self-provision and try different cloud database services. Developers connect their applications to database services through application frameworks (e.g., Quarkus, Go, Python) that leverage service binding. RHODA gives database admins and developers the ability to “Try before you buy” options and fosters experimentation and discovery of which databases fit for different workloads.

Developers reap many benefits from such easy database access. They can start work immediately because they do not need to be granted access credentials from database administrators. If properly set up, developers are presented with a curated list of DBaaS vendors and instances that they are authorized to use, and they can pick those that are most appropriate for their application.

There are also benefits for database administrators and IT Ops. They no longer have to deal with the back and forth with developers needing access credentials. They can provide access without passing around important access credentials via Slack or email.

They have more control and visibility. They can prescribe specific databases or DBaaSes that their developers can see and access. They get cluster-wide visibility into what apps are using which credentials. Additionally, RHODA provides a consistent way to work across DBaaS vendors. It lets database admins and ops employ a standardized, organized approach to managing credentials.



All of those features and capabilities help businesses bring applications to market faster. They let a company focus on business requirements and serving its users and customers. Most importantly, Red Hat OpenShift Database Access is part of a family of curated services that work cohesively to support today's cloud-native app development.

It is delivered on top of managed Red Hat OpenShift, as are many other services needed to use the growing volumes of data that businesses routinely use today. For example, RHODA complements Red Hat OpenShift API Management, which accelerates the development and deployment of API-first microservices-based applications. Similarly, it works well with Red Hat OpenShift Streams for Apache Kafka, which is a managed cloud service that reduces the operational cost and complexity of deploying real-time streaming data applications across hybrid-cloud environments. And it can be used more easily to pull data into Red Hat OpenShift Data Science, which is a cloud service that gives data scientists and developers an AI/ML platform for building intelligent applications.

## Key takeaways

The move to cloud and the explosion in data volumes present a perfect storm for many businesses today. Controlled access to databases upon which new applications are developed and services are deployed is critical to maintaining compliance, controlling costs, and more.

Traditional approaches that have all access requests funneled through and managed by database administrators are inefficient. The admins are overburdened with requests and the back-and-forth exchanges with developers.

Developers are frustrated because it takes too long to get access credentials. Such delays in fast-paced development environments are not acceptable. As a result, developers take shortcuts. They share credentials among themselves. The cumulative results when trying to use traditional access approaches in modern businesses are:

- Work is slowed
- Business units cannot react in a timely manner to new opportunities
- Redundant database use increases
- Costs are driven up
- IT operations lose control

What's needed is a modern approach to database access control. A suitable approach gives database administrators a way to standardize and simplify access to databases and DBaaS. They can provide selectable lists of available services and make those services available via a simple user interface or within their normal CI/CD tools.

Developers get a self-service model for accessing databases. That eliminates the delays and frustrations of traditional approaches. And that, in turn, allows developers to experiment with different databases more easily and quickly match the right database with their applications.

Given that much of today's modern application development centers on cloud-native approaches and the use of Kubernetes, it makes sense that a database access solution would work within that environment.

Red Hat OpenShift Database Access is such a solution. It provides the database administrators with the control and visibility into the use of their database services. It delivers the consistency IT teams need for governance. And it gives developers the needed agility via streamlined access to databases. In summary, RHODA:

- Offers easy consumption of DBaaS from ISVs like MongoDB Atlas, Crunchy Bridge, Amazon RDS, and CockroachCloud on Kubernetes clusters
- Provides easy access management, monitoring, and control by administrators of DBaaS, providing usage and status information across clusters and clouds
- Enables an end-to-end managed support experience.

Red Hat OpenShift Dedicated or Red Hat OpenShift Service on AWS users can access RHODA from the Red Hat hybrid cloud console using this link: [red.ht/dbaccess](https://red.ht/dbaccess).

Learn more about RHODA from these demos:

- [IT Ops demo video](#)
- [Developer demo video](#)



### About RTInsights

**RTInsights** is an independent, expert-driven web resource for senior business and IT enterprise professionals in vertical industries. We help our readers understand how they can transform their businesses to higher-value outcomes and new business models with AI, real-time analytics, and IoT. We provide clarity and direction amid the often-confusing array of approaches and vendor solutions. We provide our partners with a unique combination of services and deep domain expertise to improve their product marketing, lead generation, and thought leadership activity.



### About Red Hat

**Red Hat** is the world's leading provider of enterprise open-source software solutions, using a community-powered approach to deliver reliable and high-performing Linux, hybrid cloud, container, and Kubernetes technologies. Red Hat helps customers integrate new and existing IT applications, develop cloud-native applications, standardize on our industry-leading operating system, and automate, secure, and manage complex environments. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500. As a strategic partner to cloud providers, system integrators, application vendors, customers, and open-source communities, Red Hat can help organizations prepare for the digital future.

Copyright © 2022 RTInsights. All rights reserved. All other trademarks are the property of their respective companies. The information contained in this publication has been obtained from sources believed to be reliable. NACG LLC and RTInsights disclaim all warranties as to the accuracy, completeness, or adequacy of such information and shall have no liability for errors, omissions or inadequacies in such information. The information expressed herein is subject to change without notice.