

Buyers Guide: **Benefits of choosing EDB as the application database for Red Hat OpenShift**



Is this guide for you?



You are already standardized on Red Hat OpenShift



You operate across hybrid environments



You run proprietary databases



AI initiatives are on the roadmap



Availability requirements exceed 99.99%



Database teams spend too much time on patching, upgrades, and firefighting

If so, database strategy is no longer optional.

Executive snapshot

Red Hat® OpenShift® is the industry's leading hybrid cloud application platform. It is the foundation for how applications are developed, deployed, and operated across on-premises, private, and public cloud environments.

The role of the database has changed for many modern enterprise organizations.

Traditionally, databases have been treated as a special case within organizations. They were managed separately, governed differently, and usually tied to proprietary platforms. That model is increasingly out of sync with how organizations want their platforms to operate. It drives cost, limits automation, and leads to friction between application teams and the data they depend on.

In the meantime, Kubernetes has matured to support the tooling, automation, and operational patterns required to run stateful, mission-critical workloads enterprises need. As a result, database strategy has become increasingly important to platform strategy.

This buyer's guide is for technical leaders and architects wanting to make a deliberate choice about the database foundation for Red Hat OpenShift: one that avoids fragmentation, reduces long-term risk, and supports future innovation.

At a glance

- Red Hat OpenShift is a strategic application platform decision
- Database choices can directly affect cost, risk, and agility
- Fragmented or proprietary approaches no longer scale
- Kubernetes is ready for mission-critical data
- The window for ad-hoc decisions is closing as database strategy becomes inseparable from platform strategy.



The strategic question organizations must answer

If Red Hat OpenShift is your application platform of choice, what is your database platform?

This is no longer a tactical question. The database choice shapes how far automation extends, how consistent governance is, and how expensive it is to operate the platform over time.

Many organizations modernize applications first. CI/CD pipelines are standardized. Infrastructure becomes more automated. But databases are left behind. They're either running on legacy, proprietary platforms or assembled from unmanaged open source components.

Over time, this means:

- Automation stops at the database boundary
- Security and compliance control is limited
- Platform teams support multiple operating models
- Costs become harder to predict and control

The result is a partial platform strategy. Red Hat OpenShift moves quickly, but the data layer does not.

A strategic database choice should strengthen Red Hat OpenShift as a platform, not just sit alongside it. Furthermore, it should reduce exceptions by aligning databases with the same security, automation, and governance models native to Red Hat OpenShift, rather than introducing new tools and processes. And it should make future change easier, not harder, by facilitating portability, standardization, and consistent operation across environments.

The two strategic mistakes organizations keep making

As organizations try to align databases with Red Hat OpenShift, most organizations fall into one of two patterns.

Mistake 1: Treating Postgres on Red Hat OpenShift as a DIY exercise

Running community Postgres on Red Hat OpenShift can feel like the most open and flexible option. There are no Postgres licences, and full control sits with internal teams.

In practice, enterprise database operation involves far more than running a container. High availability, backups, upgrades, security, monitoring, and recovery all need to be engineered, tested, and supported.

Over time:

- Platform teams spend time maintaining database tooling instead of improving the platform
- Operational knowledge becomes fragmented and hard to scale
- Reliability depends on custom processes rather than standard patterns

The result: Postgres licence cost is replaced by hidden engineering effort, and the platform slows as complexity grows.

Mistake 2: Keeping proprietary databases alongside Red Hat OpenShift

At the other end of the spectrum, some organizations continue running proprietary databases next to Red Hat OpenShift rather than integrating the data layer into the platform.

While familiar, this approach introduces friction. Proprietary databases are not designed around Kubernetes operating models. Commercial licensing and operating models are not aligned with Kubernetes-native automation patterns, limiting how far these databases can be integrated into platform workflows. Separate infrastructure and governance models persist.

Over time:

- Costs and audit exposure increase
- Application and data teams operate differently
- Platform consistency breaks down

The result: Red Hat OpenShift becomes a hosting layer, underutilized as a true application platform.

Why both approaches fall short

Both paths treat the database as an exception, either something to be hand-built or left untouched. Neither delivers a coherent platform strategy.

A strategic database choice for Red Hat OpenShift should combine openness with enterprise-grade operational capability.



What “strategic” actually means for a database on OpenShift

Not every database that can run on Red Hat OpenShift is created equally.

A strategic choice means moving toward a Database-as-a-Service (DBaaS) model. Databases are provisioned, operated, and governed through the platform, rather than built and maintained individually by each team.

In practice, a strategic database needs to meet certain criteria:



Aligned to the Red Hat OpenShift operating model

Integrates with platform security, networking, lifecycle management, and automation.



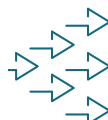
Sovereignty and portability

Control over where data lives and how it moves across hybrid environments.



Enterprise resilience by default

High availability, backup, and recovery are built in, automated, and proven.



One repeatable operating model

Consistent patterns across teams and environments, with minimal variation



Supports multiple data needs

Transactional workloads, analytics, and emerging AI use cases without sprawl.

If a database needs special handling, parallel tooling, or bespoke processes, it likely won't meet an organization's needs long-term.

Navigating the Alternatives: How EDB Postgres AI Compares on Red Hat OpenShift

When architectural teams standardize on Red Hat OpenShift, evaluating alternative data layer strategies is a critical step. While several vendors and community options are frequently considered, each introduces specific tradeoffs regarding operational overhead, licensing complexity, or architectural fit.

Below is a direct comparison of the primary alternatives your team may encounter alongside or instead of EDB Postgres AI in an OpenShift environment:

Option / Competitor	Core Architecture & Primary Use Case	Primary Strength	Critical Gaps on Red Hat OpenShift
Oracle Database	Proprietary, legacy relational database; transactional and mixed workloads.	Mature enterprise features and deep legacy application entrenchment.	Licensing complexity and high costs in containerized/cloud environments; potential for higher Total Cost of Ownership (TCO); non-native Kubernetes orchestration.
Crunchy Data	Open-source Postgres distribution and Kubernetes Operator.	Strong open-source compliance; popular in US government and defense sectors.	Lacks native Oracle compatibility layers; limited out-of-the-box analytical/AI workload unification; smaller global enterprise support scale.
Percona	Open-source, community-focused database software and support.	Price-competitiveness; community-friendly, per-instance (rather than CPU/core) licensing.	"No native cross-cluster replication orchestration; lacks automated multi-region failover tokens (higher risk of split-brain/data loss); rigid single-namespace architecture creates overhead at enterprise scale." Lacks advanced enterprise high-availability automation; no native Oracle compatibility; lower Kubernetes operator maturity for multi-cluster enterprise scale.
Snowflake	Cloud-native data warehouse; analytical and large-scale data processing.	High performance for cloud data warehousing and complex analytics.	Purely analytical architecture; not suitable for low-latency, mission-critical transactional (OLTP) workloads.
Community / Open Source Postgres	Pure, unmanaged open-source database.	Architectural freedom and extensibility; zero licensing or subscription costs.	No enterprise support SLAs; no automated high availability (HA) orchestration; heavy "Do-It-Yourself" (DIY) engineering overhead for patching and upgrades.
EDB (EnterpriseDB)	Enterprise-grade PostgreSQL platform optimized for cloud-native, hybrid-cloud, and multi-cloud infrastructure.	Native Oracle compatibility (PL/SQL support); built-in high-availability automation via CloudNativePG; enterprise-grade 24/7 SLAs; unified transactional and analytical (HTAP/Lakehouse) capabilities.	None of the architectural gaps found in community or niche variants; requires a subscription/commercial license (not \$0 upfront like pure community open-source).

"Competitive information is based on publicly available documentation and EDB's assessment as of June 2026. Competitive capabilities may vary depending on configuration, implementation, use case and/or operating environment."

Why EDB Postgres AI is a strategic database for Red Hat OpenShift

EDB Postgres AI® (EDB PG AI) is designed to meet these requirements without compromise.

It integrates with Red Hat OpenShift's operating model. This allows databases to be provisioned, secured, and operated using the same patterns as applications. High availability, automation, back-up, recovery and lifecycle management are built into the foundation.

EDB PG AI provides enterprise-grade Postgres capabilities, including compatibility features that simplify migration from legacy proprietary platforms. This allows organizations to modernize seamlessly without disruptive rewrites.

Just as importantly, EDB PG AI supports a unified data foundation. Transactional, analytical, and AI workloads are all supported within a single platform. This reduces data fragmentation and long-term total cost of ownership (TCO).

The result is a database platform that extends Red Hat OpenShift's value to any organization.



Why this matters right now

A short-term database decision can have a long-term impact on your organization's ability to operate efficiently. Several influences make this decision paramount for modern businesses:



Rising proprietary database costs and audit pressure



Platform consolidation initiatives



Increasing regulatory and data-sovereignty requirements



AI strategies depending on unified, governed data

Next steps

The best starting point isn't a wholesale migration or massive infrastructure shift. It's finding clarity.

Clarify the role you need the database to play in your Red Hat OpenShift platform. Identify where current approaches are creating costs or risk. Then evaluate how enterprise-grade Postgres strategy can support today's workloads and scale with tomorrow's growth.

Explore how EDB and Red Hat work together to deliver an enterprise-grade Postgres foundation for OpenShift.

[Get in touch with EDB today](#) for more information.



www.enterprisedb.com

About EDB Postgres AI

EDB Postgres AI is the first open, enterprise-grade sovereign data and AI platform, with a secure, compliant, and fully scalable environment, on premises and across clouds. Supported by a global partner network, EDB Postgres AI unifies transactional, analytical, and AI workloads, enabling organizations to operationalize their data and LLMs where, when, and how they need them.

© EnterpriseDB Corporation 2026. All rights reserved

