TECHNICAL BRIEF

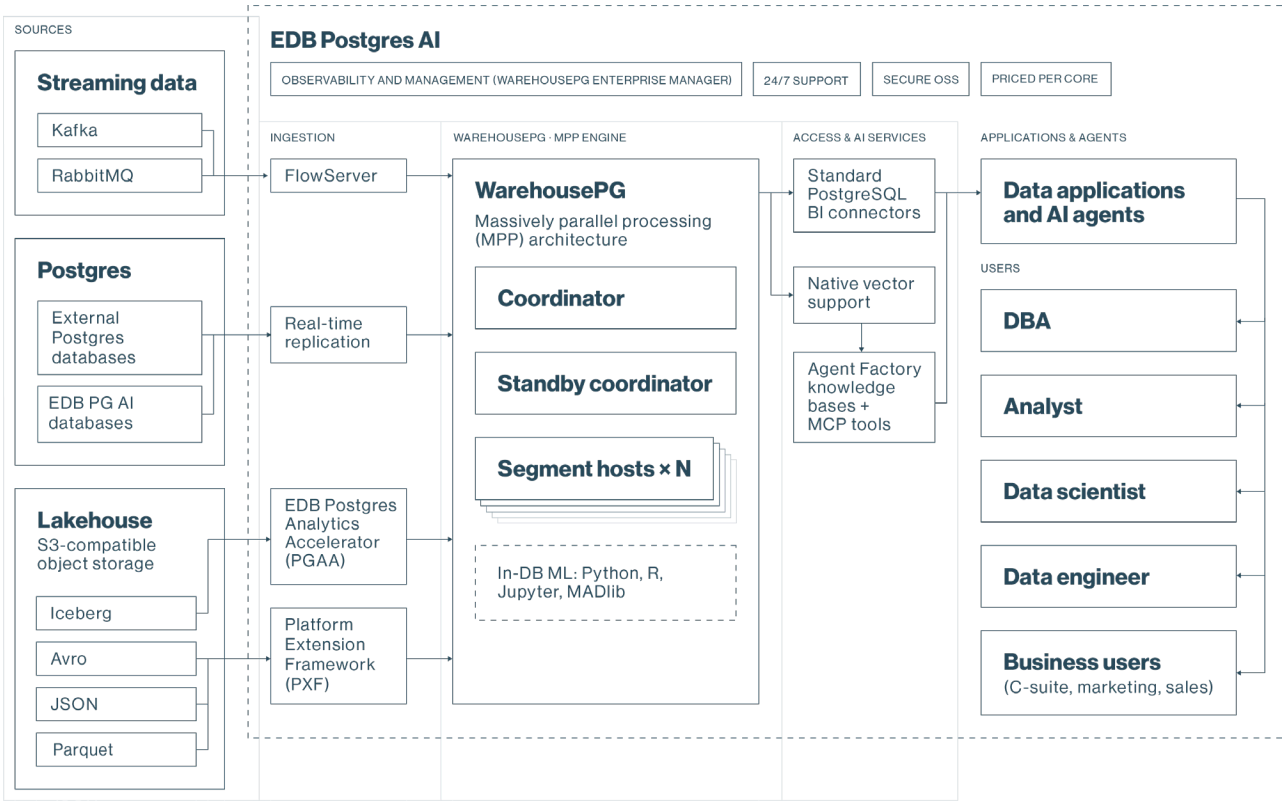
EDB Postgres® AI for WarehousePG

Architecture, performance, and enterprise deployment model

Solution overview

Warehouse Analytics is a modern data warehousing solution powered by EDB Postgres AI for WarehousePG. WarehousePG is a sovereign, Postgres-based analytical database, built on the proven open source PostgreSQL engine, that delivers petabyte-scale performance through a massively parallel processing (MPP) architecture. EDB Postgres AI surrounds and extends this foundation with enterprise capabilities covering lakehouse integration, real-time streaming ingestion, AI-readiness, and unified cluster observability—all within a predictable, core-based licensing model designed for enterprise scale.

For organizations evaluating modern data warehouse infrastructure, the open source foundation of WarehousePG provides a proven MPP engine built on Postgres, bringing the ecosystem familiarity, tooling compatibility, and SQL standards that organizations already rely on, combined with horizontal scalability and broad deployment flexibility. EDB Postgres AI adds the operational, analytical, and AI capabilities required to run that engine at enterprise scale, in regulated environments, and across distributed workload types, without introducing vendor dependency or unpredictable cost structures.



Analytics without limits. Infrastructure without surprises.

Economics and sovereign control

EDB Postgres AI for WarehousePG provides full deployment flexibility—on-premises, private cloud, or hybrid—with a per-core pricing model that eliminates the variable credit consumption structures associated with cloud-native analytics platforms. Organizations gain complete control over where data resides and exactly what the platform costs, ensuring alignment with both financial planning requirements and regulatory data residency mandates. McKnight Consulting Group found that this model enables organizations to redirect up to 58% of their cloud analytics spend back into strategic growth by eliminating variable credit burn and proprietary lock-in.

Performance at scale

The MPP architecture distributes query execution across independent compute segments to maintain consistent throughput under high concurrency. Unlike cloud-native warehouses that rely on spinning up redundant clusters to absorb peak demand, WarehousePG handles concurrent users and AI agent queries with lower performance degradation. Independent benchmark results show WarehousePG delivers up to 52% more consistent concurrency performance compared to leading cloud data warehouses, without requiring idle cluster capacity to buffer peak traffic.

Modernization and AI readiness

The platform delivers native lakehouse interoperability through the Postgres Analytics Accelerator (PGAA) extension, real-time event stream ingestion through FlowServer, and high-performance vector indexing, enabling organizations to migrate from legacy infrastructure and deploy AI workloads without introducing separate specialized systems for each capability. Organizations migrating from Greenplum have executed seamless cluster transitions in less than two hours with zero data movement.

Reliability and unified operations

WarehousePG Enterprise Manager (WEM) provides centralized observability for distributed MPP clusters, offering real-time health monitoring and query diagnostics across 12 integrated dashboards with less than 1% CPU overhead.

High availability is built directly into the WarehousePG MPP engine, featuring automatic segment failover, standby coordinator promotion, and a robust shared-nothing architecture. For larger operations, multi-cluster disaster recovery (DR) is powered by continuous WAL-based shipping. This ensures that even customers running petabyte-scale clusters can strictly meet their required recovery point objectives (RPO) and recovery time objectives (RTO).

The open source foundation

Because Postgres is the foundation, WarehousePG inherits native compatibility with the PostgreSQL wire protocol; standard SQL tooling; and the broad ecosystem of drivers, BI frameworks, and data engineering platforms that organizations already use. On top of this foundation, WarehousePG delivers a shared-nothing MPP architecture capable of petabyte-scale analytics, high query concurrency, and mixed operational and analytical workloads, thereby providing an open, flexible alternative to proprietary data warehouse platforms and cloud-only analytics services across on-premises, public cloud, and hybrid environments.

Core MPP architecture

At the core of WarehousePG is a shared-nothing MPP architecture that distributes both data and query execution across multiple compute nodes in a cluster. This architecture enables horizontal scaling by allowing each node in the cluster to process a portion of the dataset independently while coordinating execution through a central query planner. The system consists of a coordinator node and multiple worker nodes.

The coordinator node serves as the entry point for client connections and is responsible for parsing SQL queries, generating distributed execution plans, managing metadata, and orchestrating query execution across the cluster. Applications connect to the system using the standard PostgreSQL wire protocol, allowing existing tools, drivers, and applications built for PostgreSQL to interact with WarehousePG without modification.

Worker nodes form the execution layer of the cluster. Each worker node stores a portion of the distributed dataset and executes query fragments generated by the coordinator. When a query is issued, the coordinator divides the workload into parallel tasks and dispatches them to the worker nodes, where they operate on local data partitions simultaneously. The results of these operations are exchanged between nodes using a high-performance interconnect layer designed for distributed query processing. This parallel execution model enables WarehousePG to efficiently process large-scale analytical queries involving joins, aggregations, and distributed sorting operations across massive datasets.

Storage and optimization

WarehousePG leverages PostgreSQL's mature storage engine while introducing optimizations designed specifically for analytical processing, including columnar storage support, advanced compression techniques, and distributed partitioning capabilities. These features significantly reduce I/O requirements and improve query performance for analytical workloads involving large datasets.

Key technical elements and differentiators

To help architects and platform engineers understand the technical positioning of WarehousePG relative to common analytics platforms, the following table summarizes the key considerations.

Technical Capability	WarehousePG	Typical Cloud Data Warehouse	Legacy Enterprise Warehouse
Architecture model	Shared-nothing MPP cluster built on PostgreSQL	Cloud-managed distributed compute/storage	Monolithic SMP or proprietary MPP
Query execution	Distributed parallel query execution across worker nodes	Elastic compute clusters with query services	Parallel execution but often limited scalability
Data distribution	Hash distribution and replicated tables	Managed distribution with limited user control	Vendor-defined distribution models
Storage format	Columnar storage with compression	Columnar storage	Mixed row/column depending on platform
Compression	Dictionary, RLE, and delta encoding	Vendor-managed compression	Platform-specific compression
Scalability model	Horizontal node scaling across cluster	Elastic scaling tied to cloud infrastructure	Scaling often requires hardware upgrades
SQL compatibility	Full PostgreSQL SQL support	Vendor SQL dialects	Platform-specific SQL extensions
Postgres ecosystem integration	Native compatibility with PostgreSQL tools and drivers	Limited integration outside vendor ecosystem	Typically proprietary ecosystems
Deployment flexibility	On premises, cloud, hybrid	Cloud only	On premises only
Workload management	Resource queues and query prioritization	Vendor-managed resource scaling	Platform dependent
Data sovereignty	Full control of deployment location	Limited to provider regions	Full control but limited modernization
Migration path	Compatible with PostgreSQL and Greenplum workloads	Often requires query refactoring	Difficult migrations to modern platforms
Vendor lock-in	Open architecture based on PostgreSQL	High vendor dependency	High vendor dependency

*Competitive comparisons are based on publicly available information and are subject to change as vendor offerings evolve and new information is made available. All product names, trademarks, and registered trademarks are the property of their respective owners.

Architectural advantages

The combination of distributed PostgreSQL compatibility, columnar analytics optimization, and flexible deployment architecture provides several operational advantages for enterprise data platforms.

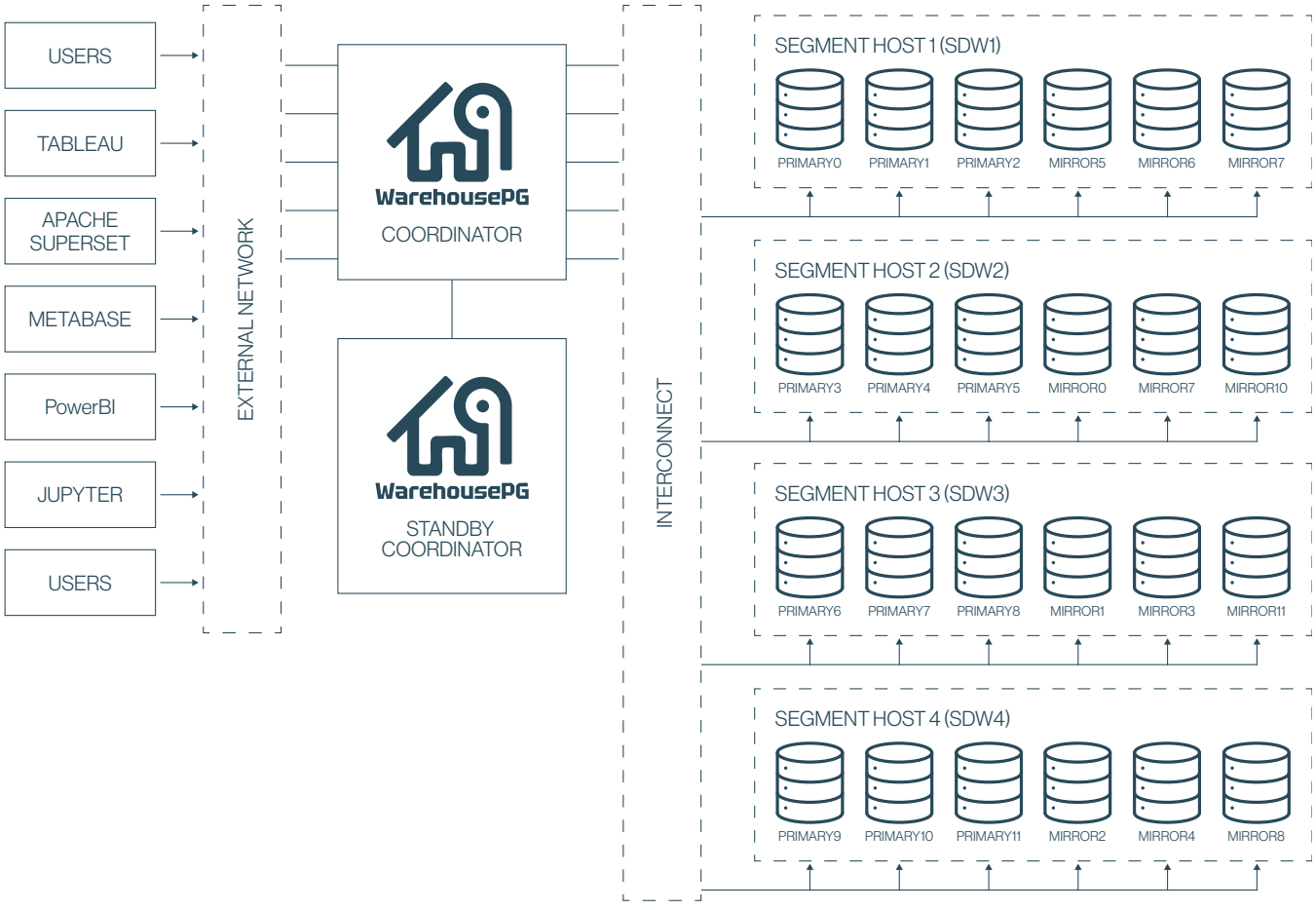
First, WarehousePG enables organizations to maintain compatibility with the rapidly expanding PostgreSQL ecosystem while gaining the benefits of massively parallel query processing. This compatibility significantly reduces the complexity of integrating analytics infrastructure with modern data engineering frameworks and business intelligence tools.

Second, the platform's shared-nothing architecture allows clusters to scale horizontally by adding worker nodes, enabling organizations to increase compute capacity and storage throughput without redesigning the underlying architecture.

Third, deployment flexibility allows organizations to align infrastructure placement with regulatory, governance, and cost requirements. Enterprises can deploy WarehousePG on premises to meet strict data sovereignty mandates, deploy in cloud environments to leverage elastic infrastructure, or operate hybrid clusters that combine both environments.

Finally, the platform's compatibility with existing PostgreSQL-based tooling and SQL workloads significantly reduces the risk associated with platform modernization projects. Organizations migrating from legacy data warehouse platforms or existing Greenplum deployments can transition workloads with minimal disruption while gaining access to modern distributed analytics capabilities.

Cluster component architecture



Cluster component breakdown

Component	Role in Architecture	Key Responsibilities	Technical Characteristics
Coordinator node	Central control plane for the cluster	Parses SQL queries, generates distributed query plans, manages metadata, coordinates execution across worker nodes	PostgreSQL-compatible endpoint, maintains system catalog and cluster metadata
Worker nodes	Distributed compute and storage layer	Executes query fragments; stores data segments; performs joins, scans, and aggregations	Parallel query execution across nodes; each node processes local data partitions
Segment processes	Execution units inside worker nodes	Process data segments during query execution and perform parallel operations	Multiple segment processes per node enable intra-node parallelism
Interconnect network	High-speed communication layer between nodes	Transfers intermediate query results, handles data redistribution during joins and aggregations	Optimized for low-latency parallel data exchange
Distributed storage layer	Physical storage of table partitions	Stores distributed table segments across cluster nodes	Supports columnar storage, compression, and partitioning
Distribution manager	Data placement and distribution control	Determines how tables are distributed across nodes (hash or replicated)	Ensures balanced data placement and efficient query locality
Query planner/ optimizer	Global query optimization engine	Converts SQL into distributed execution plans optimized for parallel execution	Cost-based optimizer with distributed execution awareness
Workload management system	Resource allocation and query prioritization	Controls concurrency, assigns resources, manages query queues	Supports workload isolation and performance governance
Replication and mirroring layer	High availability and fault tolerance	Replicates data segments across nodes to protect against failures	Enables node failover and cluster recovery
Backup and recovery system	Data protection and disaster recovery	Supports backups, WAL archiving, and restore operations	Compatible with PostgreSQL backup mechanisms
External integration layer	Ecosystem connectivity	Connects BI tools, ETL frameworks, streaming pipelines, and analytics engines	PostgreSQL wire protocol compatibility

Example logical cluster layout

Layer	Cluster Elements
Client access layer	BI tools, analytics applications, ETL pipelines
Coordinator layer	Query planner, metadata services, connection management
Layer	**Cluster Elements**
Distributed execution layer	Worker nodes running segment processes
Storage layer	Distributed columnar data segments
Networking layer	Interconnect for distributed query operations

Technical advantages of this architecture

This architecture provides several advantages for enterprise analytics environments:

Horizontal scalability

Clusters can expand by adding worker nodes without redesigning the system architecture. Each additional node contributes both compute capacity and storage throughput proportionally to the cluster's overall capability.

Parallel query execution

Large analytical queries are processed simultaneously across multiple nodes, reducing execution time. The distributed query planner decomposes complex SQL operations—including multi-table joins, window functions, and aggregations over large datasets—into parallel execution plans that run concurrently across all worker nodes.

Deployment flexibility

Clusters can run on physical infrastructure, virtual machines, cloud environments, or container platforms. This deployment model allows organizations to align infrastructure with regulatory, cost, and operational requirements without changing the platform architecture.

PostgreSQL ecosystem compatibility

Applications and tools designed for PostgreSQL can interact with WarehousePG without specialized drivers or SQL dialect changes. This compatibility encompasses standard PostgreSQL client libraries, JDBC and ODBC drivers, BI tools, ETL frameworks, and data engineering toolchains built around the PostgreSQL wire protocol.

Operational resilience

Data replication and segment mirroring protect against node failures and support enterprise disaster recovery strategies. The mirroring layer maintains synchronous copies of each data segment on separate worker nodes, enabling automatic recovery from hardware failures without manual intervention.

EDB Postgres AI platform enhancements

The open source WarehousePG engine provides the MPP foundation. EDB Postgres AI extends that foundation with a set of commercial capabilities that address enterprise requirements for lakehouse integration, real-time data ingestion, AI workload support, cluster observability, and high availability. Each enhancement is purpose-built for the distributed MPP architecture and integrates directly with the WarehousePG execution model rather than requiring external systems or data movement.

Postgres Analytics Accelerator (PGAA)

The Postgres Analytics Accelerator (PGAA) extension enables WarehousePG to query data stored in open lakehouse formats—including Apache Iceberg—at high performance, without requiring data movement into the warehouse. PGAA bridges the architectural gap between the transactional warehouse and the data lake, enabling organizations to execute MPP-scale analytical queries against external object storage as though the data were resident in the cluster.

FlowServer: Real-time streaming ingestion

FlowServer is a dedicated streaming ingestion component that enables high-volume, low-latency loading of event data directly into WarehousePG from messaging and streaming infrastructure. It is designed to handle the throughput and delivery guarantees required for production analytics on live event streams, without routing data through intermediate staging systems or general-purpose ETL pipelines.

Vector support and in-database ML

EDB Postgres AI for WarehousePG includes native support for high-performance vector operations within the MPP cluster, enabling organizations to deploy AI and machine learning workloads directly against warehouse-scale datasets without maintaining separate vector databases or moving data to external model-serving systems.

The platform includes MADlib, an open source library of in-database statistical analysis and machine learning algorithms that execute as distributed SQL functions within the WarehousePG query engine. MADlib supports a range of analytical workloads including regression analysis, classification, clustering, dimensionality reduction, and text analytics. For teams preferring Python-based workflows, the platform supports PL/Python, enabling data scientists to execute Python scripts—including calls to external ML frameworks—within the database session and against distributed data without requiring ETL to an external compute environment.

WarehousePG Enterprise Manager

WarehousePG Enterprise Manager is a centralized management and observability platform built specifically for distributed MPP clusters. It provides operations teams with a consolidated view of cluster health, query diagnostics, workload activity, and system resource utilization, replacing the need to aggregate data from per-node monitoring tools or navigate fragmented administrative interfaces.

High availability and disaster recovery

EDB Postgres AI for WarehousePG includes a layered approach to availability and recovery that addresses both local hardware failures and site-level disaster scenarios. The combination of automatic segment failover and continuous WAL-based disaster recovery provides a resilience architecture appropriate for mission-critical analytical workloads. Organizations in regulated industries with strict business continuity mandates can demonstrate defined recovery time and recovery point commitments without deploying third-party replication appliances or introducing additional vendor dependencies into the platform stack.

Feature summary

The following tables summarize capabilities available in open source WarehousePG alongside the additional capabilities introduced by EDB Postgres AI, followed by an updated architectural comparison across major analytics platforms incorporating the full EDB Postgres AI feature set.

Open source vs. EDB Postgres AI capability comparison

Capability	Open Source WarehousePG	EDB Postgres AI for WarehousePG
MPP architecture (shared-nothing, parallel query execution)	✓	✓
Columnar storage and compression	✓	✓
Hash distribution and replicated tables	✓	✓
PostgreSQL wire protocol compatibility	✓	✓
PXF (Platform Extension Framework — external data access)	✓	✓
Workload management (resource queues, query prioritization)	✓	✓
Horizontal node scaling	✓	✓
On-premises, cloud, and hybrid deployment	✓	✓
Segment mirroring and automatic failover	✓	✓
Standby coordinator	✓	✓
pgvector support	✓	✓
In-database machine learning (MADlib, PL/Python)	✓	✓
Backup and restore	✓	✓
Postgres Analytics Accelerator (PGAA) — native Apache Iceberg integration	—	✓
FlowServer (real-time streaming ingestion)	—	✓
WarehousePG Enterprise Manager	—	✓
Disaster recovery (whpg-dr, RTO < 1 hour)	—	✓
Apache NiFi connector (data flow and pipeline orchestration)	—	✓
Per-core subscription pricing	—	✓
Direct EDB engineering and migration support	—	✓

*Competitive comparisons are based on publicly available information and are subject to change as vendor offerings evolve and new information is made available. All product names, trademarks, and registered trademarks are the property of their respective owners.

Platform comparison: EDB Postgres AI vs. cloud analytics platforms

The previous table shows what EDB Postgres AI adds at the feature level. This table takes that full capability set and positions it directly against the modern cloud analytics platforms organizations most commonly evaluate—Snowflake, Amazon Redshift, and Databricks—across the technical dimensions that drive infrastructure decisions.

Technical Dimension	EDB Postgres AI for WarehousePG	Snowflake	Amazon Redshift	Databricks (Lakehouse)
Core architecture	Shared-nothing MPP built on PostgreSQL with enterprise enhancements	Multi-cluster shared data architecture	MPP cluster with managed compute nodes	Distributed compute engine over data lake
Storage model	Distributed columnar storage; native Apache Iceberg lakehouse integration via PGAA extension	Centralized storage layer with compute clusters	Columnar storage in cluster nodes	Object storage (Parquet/Delta Lake)
Compute model	Parallel query execution across worker nodes; real-time streaming ingestion via FlowServer	Elastic compute warehouses	Elastic cluster nodes	Spark-based distributed compute
Query engine	PostgreSQL-based distributed SQL with PGAA acceleration for lakehouse queries	Proprietary SQL engine	PostgreSQL-derived MPP engine	Spark SQL engine
Lakehouse integration	Native Apache Iceberg support via PGAA extension; predicate pushdown and partition pruning without ETL	Native Iceberg support (Polaris Catalog)	Redshift Spectrum with S3; limited native Iceberg	Native (primary design target)
Streaming ingestion	FlowServer: native Kafka and RabbitMQ ingestion with direct segment writes	Snowpipe (managed, serverless)	Kinesis integration; streaming COPY	Structured Streaming (Spark)
Vector and AI support	Native support for pgvector and in-database ML via MADlib	Limited native vector support	pgvector available; limited in-database ML	MLflow integration; Vector Search (managed service)
Cluster observability	WarehousePG Enterprise Manager: 12 dashboards, <1% CPU overhead, query-level distributed execution diagnostics	Query profile; cloud partner integrations required for full visibility	Performance insights; CloudWatch integration	Databricks UI; external tooling commonly required
Concurrency scaling	Parallel execution across cluster nodes; up to 52% more consistent than leading cloud data warehouses	Independent compute clusters, concurrency scaling via multi-cluster	Concurrency scaling SQL warehouses	Job clusters or SQL warehouses
Workload types	Enterprise DW, BI, real-time streaming analytics, AI/ML, vector search, agentic workloads	Analytics and BI workloads	Analytics and reporting	Analytics, ML pipelines, data engineering
SQL compatibility	Full PostgreSQL SQL; high parity with Greenplum workloads	Proprietary SQL extensions	PostgreSQL-like dialect	Spark SQL
Open architecture	Open source PostgreSQL foundation with enterprise commercial layer	Proprietary platform	Proprietary AWS platform	Open data formats; proprietary platform services
Deployment model	On-premises, cloud, and hybrid environments; sovereign deployment anywhere	Cloud-only SaaS	AWS only	Cloud-only platform
Data sovereignty control	Full control, deploy to meet any residency mandate	Limited to cloud provider regions	Limited to AWS regions	Limited to cloud provider regions
High availability	Automatic segment failover via mirror segments; standby coordinator for cluster-level resilience	Platform-managed; multi-AZ	Platform-managed; multi-AZ	Platform-managed

*Competitive comparisons are based on publicly available information and are subject to change as vendor offerings evolve and new information is made available. All product names, trademarks, and registered trademarks are the property of their respective owners.

Technical Dimension	EDB Postgres AI for WarehousePG	Snowflake	Amazon Redshift	Databricks (Lakehouse)
Disaster recovery	whpg-dr: continuous WAL shipping to warm standby; RTO < 1 hour for petabyte-scale clusters	Business Critical Edition; point-in-time restore	Automated snapshots; cross-region restore	Delta Lake time travel; limited formal DR options
Migration complexity	Minimal for PostgreSQL and Greenplum workloads; direct EDB migration support	Query refactoring often required	Moderate migration effort	Often requires major pipeline changes
Vendor lock-in risk	Low (open architecture, PostgreSQL compatible)	High	High	Moderate (open data formats but platform dependent)
Cost model	Per-core subscription; predictable flat-fee capacity pricing	Consumption-based compute pricing	Consumption-based compute pricing	Consumption-based compute and storage pricing

*Competitive comparisons are based on publicly available information and are subject to change as vendor offerings evolve and new information is made available. All product names, trademarks, and registered trademarks are the property of their respective owners.

EDB Postgres AI: The sovereign data and AI platform for the agentic enterprise

EDB PG AI brings together a unified data layer, governance, sovereign control and orchestration, and an agent runtime environment, giving enterprises a trusted foundation for AI on infrastructure they own and control. The platform unifies transactional, analytical, and AI workloads in a single Postgres-based architecture—eliminating ETL, data movement, and operational fragmentation. And you choose where and how to deploy: on-premises, cloud, managed, or certified appliance.

The outcome: production-ready sovereign AI in days or weeks, not months.

EDB POSTGRES AI

EDB Postgres® AI (EDB PG AI) is the sovereign data and AI platform for the agentic enterprise. Built on Postgres, the world's leading open source database, EDB PG AI unifies transactional, analytical, and AI workloads in a single governed architecture, on-premises and across clouds. To learn more, visit www.enterprisedb.com.