

# Exploration de l'écosystème applicatif de PostgreSQL

Note de veille technologique

Cyprien PIERRE 

2025-04-10

## Résumé

Abstract

**Mots clés :** Mots clés

## Introduction

Contexte

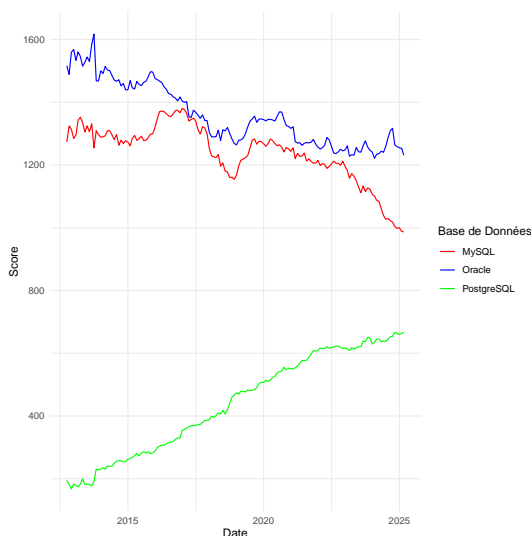


FIGURE 1 – Tendances DB-Engine Ranking pour Oracle, MySQL, et PostgreSQL

Objectif

Enjeux

Méthodologie

Concentration sur les projets les plus populaires, sourcés via PostgreSQL is Enough et 1000+ PostgreSQL Extensions, filtré à plus de 100 étoiles et dont les développements sont toujours en cours (Mars 2025).

## Fondamentaux techniques

### Extensions multi-modèles

Bien que PostgreSQL soit initialement une base de données relationnelle orientée objet, il permet de créer des types de données personnalisés pour en étendre ses fonctionnalités.

Associé à ses capacités d'extensions, cette capacité de modification du logiciel lui permet de prendre en charge la quasi totalité des modèles de données existantes comme à venir.

Un grand nombre d'extensions communautaires, parfois portées par des entreprises, sont employés à cet effet.

### Relationnel étendu

**pg\_auto\_dw** An auto data warehouse extension for Postgres

**pgtt** extension to create, manage and use Oracle-style Global Temporary Tables and the others RDBMS.

## Traitement en temps réel

TABLE 1 – Ingestion des données en temps réel

Nom	Particularité
TimescaleDB	Hypertables Chunking Compression TSPC
pg_timeseries	time-series tables
PipelineDB	<b>Discontinued</b> Streaming

**temporal\_tables** Temporal Tables PostgreSQL Extension  
**periods** recreates the behavior defined in SQL:2016 (originally in SQL:2011) around periods and tables with SYSTEM VERSIONING.

TABLE 2 – Execution programmée de scripts

Nom	Particularité
pg_timetable	Plannificateur d'évènement
pg_cron	Execution periodique
pg_later	Préexécution
bg_mon	Moniteur d'arrière plan

TABLE 3 – Capture et traitement d'évènements dans la base de données

Nom	Particularité
wal2json	Changeset extraction JSON output
walex	CDC in Elixir

## Géospatial

PostGIS est l'une des extensions de PostgreSQL les plus connues. Elle permet principalement de prendre en charge des modèles de données géographiques, géométriques et géospaciales.

TABLE 4 – Outils de manipulation de données géospaciales

Nom	Particularité
pgRouting	Routage géospatial Analyse de réseau
MobilityDB	Analyse et de gestion des trajectoires
h3-pg	bindings for Uber's H3
pgpointcloud	Storing LIDAR data
giswater_dbmodel	SQL files for Giswater's database model



FIGURE 2 – Logos : PostGIS, pgRouting, MobilityDB, H3, GISWater

**pg\_svg** Postgres/PostGIS functions for generating SVG documents usage : génération de cartes, de plans, etc.  
 Raster format, Bitmap extensions etc.

## Données non structurées

**Apache AGE** Graph database optimized for fast analysis and real-time data processing. It is provided as an extension to PostgreSQL (OpenCypher)

**MS DocumentDB** offers a native implementation of document-oriented NoSQL database, enabling seamless CRUD opera-

tions on BSON data types within a PostgreSQL framework

**Marten** .NET Transactional Document DB and Event Store on PostgreSQL



FIGURE 3 – Logos : Apache AGE, FerretDB, pg\_duck, Marten, Pongo

**JsQuery** json query language with GIN indexing support

**FerretDB** A proxy that converts MongoDB 5.0+ wire protocol queries to SQL and uses PostgreSQL with DocumentDB extension as a database engine

**dox** A Document Database API extension for Postgres

Expliquer le principe "NoSQL" et les divergences d'interprétations

**pg\_render** Render HTML in SQL. PostgreSQL extension

**pgpdf** pdf type for Postgres

**Pongo** Mongo but on Postgres and with strong consistency benefits

Pour la mise en cache Solutions : pg\_catcache (LRU), pgmemcache (intégration Memcached)

Mise en cache et file d'attente : postgres's Unlogged table puis memcache (ou equivalent)

**pg\_jsonschema** PostgreSQL extension providing JSON Schema validation

**pg\_duckdb** a Postgres extension that embeds DuckDB's columnar-vectorized analytics engine and features into Postgres

## Récupération d'information et IA

*Dans un monde plus ouvert qu'il n'a jamais été, notre Société traverse une nouvelle mutation en passant de l'économie industrielle à l'économie du service et de la connaissance. – Urban-Galindo<sup>1</sup>*

La mise en place d'un environnement d'apprentissage, d'exploration et de partage de connaissance offre la capacité de développer

son expertise.<sup>2</sup> Il s'agit d'une opportunité stratégique de long terme permettant le maintien des connaissances au sein d'une entreprise dans un contexte de raréfaction des ressources humaines,<sup>3</sup> lutant ainsi contre l'amnésie organisationnelle.<sup>4</sup>

Discuter de l'avantage stratégique de pgvector face à des bases dédiées (Elasticsearch).

**pgVector** Implémentation des vecteurs

**pgvectorscale** A complement to pgvector for high performance, cost efficient vector search on large workloads. (timescaledb)

**pg\_vectorize** A Postgres extension that automates the transformation and orchestration of text to embeddings and provides hooks into the most popular LLMs. This allows you to do vector search and build LLM applications on existing data with as little as two function calls.

**lantern** PostgreSQL vector database extension for building AI applications



FIGURE 4 – Logos : ParadeDB, ZomboDB, PGroonga, PostgresML, Lantern

Dans le paysage en évolution de l'IA, la mal-léabilité de PostgreSQL devient un avantage stratégique. Le framework PostgresML, intégrant des modules tels que pgCat, pgVector et pg\_partman, permet d'architecturer directement au niveau de la base de données des systèmes de Machine Learning.

**PostgresML** Postgres with GPUs for ML/AI apps

**pgai** A suite of tools to develop RAG, semantic search, and other AI applications more easily with PostgreSQL (TimescaleDB)

Cas concret : Chatbot RAG avec pgvector + Llama2

**pgvecto.rs** Scalable, Low-latency and Hybrid-enabled Vector Search in Postgres. Revolutionize Vector Search, not Database

**pg\_auto\_embeddings** Text embeddings calculation for Postgres, without extensions. Simple, atomic, supports Ope-

nAI/Anthropic models. Does not require any additional extensions, making it suitable for managed databases and other restricted environments

Korvus, Ollama...

**datasketches-postgresql** extension providing approximate algorithms based on Apache Datasketches-cpp

**pgfaceting** extension to quickly calculate facet counts using inverted index built with roaring bitmaps

**TopN** Extension that returns the top values in a database according to some criteria

**OneSparse** Accelerated Sparse Linear Algebra with Postgres and SuiteSparse

**ParadeDB** (recherche sémantique)

**ZomboDB** Making Postgres and Elasticsearch work together

**zhparser** zhparser is a PostgreSQL extension for full-text search of Chinese language

**pgroonga** extension to use Groonga as index. PGroonga makes PostgreSQL fast full text search platform for all languages!

**pg\_bigm** The pg\_bigm module provides full text search capability in PostgreSQL. This module allows a user to create 2-gram (bigram) index for faster full text search.

**smlar** Extension for an effective similarity search

**pg\_roaringbitmap** Roaring bitmaps are compressed bitmaps which tend to outperform conventional compressed bitmaps such as WAH, EWAH or Concise. In some instances, roaring bitmaps can be hundreds of times faster and they often offer significantly better compression. They can even be faster than uncompressed bitmaps

## Solutions d'optimisation

- Fonctions natives
- Cas d'usage : Transactions ACID à haute charge (ERP, systèmes financiers)

OLTP historiquement "lent" comparé à OLAP

**Aqo** Adaptive query optimization for PostgreSQL

**E-Maj** extension that tracks updates on tables sets with rollback capabilities

**pg\_partman** Partition management extension for PostgreSQL

**pg\_pathman** Partitioning tool for PostgreSQL

**pgslice** Postgres partitioning as easy as pie  
Citius ajoute la capacité de mise à l'échelle horizontale par répartition de la charge de calcul sur les serveurs disponibles. Cette approche est appelée *sharding*. Cette extension ajoute notamment la prise en charge des tables distribuées, des tables de références, du stockage en colonnes (Wide column store) et intègre un moteur de requête distribué.

TimescaleDB propose une approche différente appelée *chunking*.

**PG-Strom** Extension conçue pour accélérer les charges de travaux en lots et les traitements analytiques en les exécutant lorsque nécessaire sur les GPU disponibles. (from HeteroDB)

**pgtune** tuning PostgreSQL config by your hardware

## Stratégies de stockage

**zheap** (réduction WAL) Extension initialement créée par EDB puis maintenu par Cyber-tec avant d'être intégré à PostgreSQL core en version<sup>5</sup>

**pg\_dump** (algo GZip/LZ4/Zstd)

How did TimescaleDB comp is working?

Alternatives to CLUSTER and VACUUM

**pg\_squeeze** Optimise la mémoire en réorganisant les tables et indexes en minimisant les verrouillages de la base de données (Logical decoding on server only using background workers)

**pg\_repack** Optimise la mémoire en réorganisant les tables et indexes en minimisant les verrouillages de la base de données (Triggers on Client + Server)

**pg\_parquet** Copy to/from Parquet in S3 or Azure Blob Storage from within PostgreSQL

**pg\_tier** Postgres Extension written in Rust, to enable data tiering to AWS S3

**pgwarehouse** Easily sync your Postgres database to a Snowflake, ClickHouse, or DuckDB warehouse

**storage** S3 compatible object storage service that stores metadata in Postgres

**pg\_incremental** Incremental Data Processing in PostgreSQL

**pg\_ivm** Incremental View Maintenance implementation as a PostgreSQL extension

**pgpq** Stream Apache Arrow RecordBatches into Postgres

**VOPS** VOPS provides set of standard operators for tile types, allowing to write SQL queries in the way similar with normal SQL queries.

**pgsync** Sync data from one Postgres database to another

**OrioleDB** Remplace le moteur de stockage dans le heap de PostgreSQL, change de row à column based querying, améliore la compression des données, nettoie automatiquement les espaces libres.

**Citus** (Microsoft) : Ajoute la capacité de mise à l'échelle horizontale par répartition de la charge de calcul sur les serveurs disponibles. Cette extension ajoute notamment la prise en charge des tables distribuées, des tables de références, du stockage en colonnes (Wide column store) et intègre un moteur de requête distribué.

**columnar** Postgres-native columnar storage extension

## Indexation avancée

**BRIN** (Big Data),

**GIN** (full-text),

**Bloom** (requêtes complexes)

**dexter** Automatic indexer for Postgres

**RUM** RUM access method - inverted index with additional information in posting lists

**HypoPG** Création d'index hypothétique

*An hypothetical, or virtual, index is an index that doesn't really exists, and thus doesn't cost CPU, disk or any resource to create. They're use-*

*ful to know if specific indexes can increase performance for problematic queries, since you can know if PostgreSQL will use these indexes or not without having to spend resources to create them. – HypoPG*

**index\_advisor** Outils pour recommander des index appropriés

**pg\_hint\_plan** Extension adding support for optimizer hints in PostgreSQL

**pg\_plan\_advsr** extension for automated execution plan tuning

## Architectures distribuées

**ElectricSQL** (CRDTs)

**PGLite** Embeddable Postgres with real-time, reactive bindings. (Wasm/Deno)

Use case : Synchronisation offline-first (apps mobiles)

Expliquer les intérêts en matière de performance et de fort découplage des environnements serveurs et clients.

Étudier les garanties de pglogical en environnement distribué

**pg\_replicate** a Rust crate to quickly build replication solutions for Postgres.

**peerdb** Fast, Simple and a cost effective tool to replicate data from Postgres to Data Warehouses, Queues and Storage

**pg\_failover\_slots** PG Failover Slots is for anyone with Logical Replication Slots on Postgres databases that are also part of a Physical Streaming Replication architecture.

**pglogical** provides logical streaming replication for PostgreSQL, using a publish/subscribe model

**repmgr** A lightweight replication manager for PostgreSQL

## Écosystème applicatif

### Compatibilité

**Compatibilité**, n.f. Le fait pour un matériel ou un logiciel d'être utilisable en connexion avec d'autres

*matériels, d'autres logiciels, dans un système informatique défini, sans en altérer les conditions de fonctionnement.* – CNRTL

La compatibilité diffère de l'interopérabilité par l'absence de nécessité de connaître l'architecture interne des solutions informatiques avec lesquelles le système étudié s'interface.

**pg\_net** A PostgreSQL extension that enables asynchronous (non-blocking) HTTP/HTTPS requests with SQL

**pgsql-http** Client HTTP permettant de récupérer des pages Web depuis la base de données

API REST :

**prest** PostgreSQL, REST, low-code, simplify and accelerate development, instant, realtime, high-performance on any Postgres application, existing or new

**PostgREST** API RESTful gérée par PostgreSQL

**postgres-meta** A RESTful API for managing your Postgres. Fetch tables, add roles, and run queries

API GraphQL :

**PostGraphile** API GraphQL bidirectionnelle gérée par PostgreSQL I/O

**pg\_graphql** Permet de requêter PostgreSQL avec une API GraphQL

**graphql-engine** Blazing fast, instant realtime GraphQL APIs on all your data with fine grained access control, also trigger webhooks on database events

WebHook :

**pgstream** Change Data Capture CLI et librairie pour PostgreSQL avec un envoi via WebHook

WebSocket :

**postgres-websockets** Middleware permettant d'ajouter la prise en charge de Websockets à PostgreSQL en l'interfaçant avec les commandes LISTEN/NOTIFY

## Interopérabilité

**Interopérabilité**, n.f. *Capacité que possède un produit ou un système, dont les interfaces sont intégralement connues, à fonction-*

*ner avec d'autres produits ou systèmes existants ou futurs et ce sans restriction d'accès ou de mise en œuvre.* – Wikitionnaire

Évaluer les performances des FDW (ex. mysql\_fdw vs oracle\_fdw) et leur utilité dans les architectures hybrides.

postgres\_fdw

**wrappers** Postgres Foreign Data Wrapper development framework in Rust.

**multicorn2**

How it works??

**duckdb\_fdw** Permet de requêter DuckDB

**hdfs\_fdw** Permet de requêter Apache Hadoop

**mongo\_fdw** Permet de requêter MongoDB

**mysql\_fdw** Permet de requêter MySQL

**oracle\_fdw** Permet de requêter Oracle

**parquet\_fdw** Permet de requêter Apache Parquet

**ogr\_fdw** Permet de requêter OGR, un sous-ensemble de GDAL

**redis\_fdw** Permet de requêter Redis

**sqlite\_fdw** Permet de requêter SQLite

**tds\_fdw** Permet de requêter les bases de données utilisant le protocole TDS comme MS SQL Server

**kafka\_fdw** kafka foreign database wrapper for postgresql

**wrappers** Postgres Foreign Data Wrapper development framework in Rust

Discuter des défis techniques (ex. différences de syntaxe SQL) et des outils comme Babelfish pour réduire la dépendance à Microsoft SQL Server.

**SynchDB** Extension permettant la migration de données d'un DBMS externe vers PostgreSQL

**babelfish\_extensions** Babelfish for PostgreSQL provides the capability for PostgreSQL to work with applications written for Microsoft SQL Server. Babelfish understands the SQL Server wire-protocol and T-SQL, the Microsoft SQL Server query and procedural language, so you don't have to switch database drivers or rewrite all of your application queries

**orafce** The "orafce" project implements in

Postgres some of the functions from the Oracle database that are missing (or behaving differently). Those functions were verified on Oracle 10g, and the module is useful for production work.

**pgroll** PostgreSQL zero-downtime migrations made easy

**pgCompare** a straightforward utility crafted to simplify the data comparison process, providing a robust solution for comparing data across various database platforms

**pg-schema-diff** Go library for diffing Postgres schemas and generating SQL migrations

## Écosystème de développement

TABLE 5 – Interfaces et clients

Nom	Particularité
psql	CLI Solution native
pgcli	CLI
pghero	Performance dashboard
pgwatch	Metrics monitor/dashboard
rainfrog	TUI
mathesar	Spreadsheet-like Collaborative User-friendly

**plpgsql\_check** Linter dédié à l'analyse des procédures et fonctions PL/pgSQL

Framework de développement :

**pg\_tle** Framework for building trusted language extensions for PostgreSQL

**pgrx** Build Postgres Extensions with Rust

Officiels : PL/pgSQL, PL/Python, PL/Perl, PL/Tcl

Communautaires : PL/Dotnet pour les langages utilisant l'environnement .NET (C#, F#, VB...), PL/Haskell, PL/Java pour les langages utilisant la JVM (Java, Scala, Clojure...), PL/V8 pour les langages pouvant fonctionner sur la machine virtuelle V8 (JS, TS, AS, WASM...), PL/sh pour exécuter des scripts d'administration systèmes (BASH, ZSH, FISH...) et PL/R pour créer des pipelines ETL et automatiser la génération de graphiques.

PL/Rust

D'autres ont été créées par le passé mais sont actuellement non maintenues. Nous pouvons identifier PL/Julia, PL/Lua, PL/PHP, PL/Ruby et PL/Scheme.

Nous pouvons réaliser quelques observations au sujet de ces langages d'extensions :

- La montée progressive de l'utilisation de Julia en science des données et son positionnement comme langage de programmation dynamique et performant pourrait entraîner un regain d'intérêt dans le développement de PL/Julia par la communauté. Son utilisation pourrait permettre de progressivement remplacer PL/R.
- De son côté, Lua étant un langage de script et d'extension, il concurrence directement Python dans cet emploi. Ce second étant très largement supporté et employé par la communauté des sciences des données et du développement en général, il est peu probable de voir PL/Lua supplanter PL/Python à l'avenir.
- PL/PHP pourrait revenir sur le devant de la scène à travers le développement d'initiatives telles que White Label Press cherchant à fournir une alternative moderne à WordPress.
- PL/XSLT,
- PL/prql,
- PL/Proxy Permet de faire des appels de procédure à distance entre les bases de données

TABLE 6 – Gestionnaires de paquets

Nom	Particularité
Pex	
pgxman	
dbdev	
trunk	

TABLE 7 – Connecteurs et ORM

Nom	Particularité
node-postgres	Client pour Node.js
postgres.js	Client pour Node.js, Deno, Bun and CloudFlare Utilisé par TypeORM et Drizzle
slonik	Client pour Node.js Type safety Composable SQL
npgsql	.NET data provider
rust-postgres	Native driver for Rust

**SQLPage** Fast SQL-only data application builder. Automatically build a UI on top of SQL queries

**pgtyped** Typesafe SQL in TypeScript

**zapatos** Zero-abstraction Postgres for TypeScript : a non-ORM database library

**pg-introspection** A strongly-typed PostgreSQL introspection library for PostgreSQL built automatically from the PostgreSQL system catalog documentation, with the TypeScript documentation for each attribute/type also pulled from the PostgreSQL documentation for easy reference directly in your editor.

**postgres-lsp** A collection of language tools and a Language Server Protocol (LSP) implementation for Postgres, focusing on developer experience and reliable SQL tooling.

**pg-sql2** Create highly dynamic SQL in a powerful and flexible manner without opening yourself to SQL injection attacks.

**plmustache** Logic-less templates for Postgres

## Gouvernance et sécurité

### Conformité et audit

**pg-safeupdate** a simple extension to PostgreSQL that raises an error if UPDATE and DELETE are executed without specifying conditions. This extension was initially designed to protect data from accidental obliteration of data that is writable by PostgREST.

**pgspot** pgspot checks extension scripts for following PostgreSQL security best practices. In addition to checking extension scripts it can also be used to check security definer functions or any other PostgreSQL SQL code.



TABLE 8 – Tests et qualité

Nom	Particularité
pgTAP	Suite applicative pour créer des tests unitaires
pgmock	In-memory PostgreSQL mock server for unit and E2E tests
pgtestdb	Quickly run tests in their own temporary, isolated, databases
bytebase	DevSecOps solution for Developer, Security, DBA and Platform Engineering teams
database-lab-engine	Enables database branching and thunder thin cloning for any database and empowers DB testing in CI/CD

TABLE 9 – Observabilité

Nom	Particularité
pgaudit	SOC2
pgsentinel	Active session history
pg_analytics	Requêter des stockages comme AWS des tables comme Iceberg ou Delta Les réponses sont envoyés à DuckDB Postgres historic workload reports
pg_developer_file	
pg_analytics	Collecte des statistiques sur les prédicats et aide à trouver les indices manquants Requêter le moniteur d'avancement
pg_query_state	
pg_stat_kcache	Regroupe les statistiques sur les lectures et écritures effectués par le système de fichiers
pg_stat_monitor	Analyse des performances
pg_wait_sampling	Sampling based statistics of wait events
PoWA	Workload Analyzer
StatsMgr	Gestion des statistiques de performance et d'utilisation
system_stats	Expose les données du système comme l'utilisation du CPU, de la mémoire et des disques Server activity monitoring
pg_activity	
pg_catcheck	Outil pour diagnostiquer la corruption du catalogue système
pg_dirtyread	Read dead but unvacuumed tuples from a PostgreSQL relation
pg_show_plans	Show query plans of all currently running SQL statements
pgassistant	Tool designed to help understand and optimize performances

La longue liste de modules et extensions d'observabilité peut indiquer une vigilance particulière des utilisateurs au fonctionnement et aux performances du DBMS.

Cet intérêt peut être une des raisons de l'amélioration progressive des performances natives et du développement d'extensions sectorielles.

## pgMustard

### Chiffrement

**pg\_anonymize** (GDPR),  
**pgcrypto** (AES-256/Q-Safe),  
**pgsodium** Modern cryptography for PostgreSQL using libsodium  
**PostgreSQL Anonymizer** Anonymization & Data Masking for PostgreSQL  
**supautils** PostgreSQL extension that secures a cluster on a cloud environment  
**walrus** Applying Row-Level-Security to PostgreSQL WAL  
Industrie : PostgreSQL Quantum-Safe (NIST PQC integration) Explorer les initiatives émergentes (ex. intégration NIST PQC). Parler de l'avancée de Google avec le craque du SHA

### Gestion des accès

**acl** Access Control Lists (ACL) PostgreSQL Extension  
LDAP/Kerberos : Intégration Active Directory native  
**pg\_iam** (PoC CNRS [source 29])  
OAuth2/OIDC : Extension  
**pg\_permissions** A simple set of views to see ALL permissions in a PostgreSQL database

### Continuité de service

TABLE 10 – Solutions de sauvegarde et de récupération

Nom	Particularité
barman	
pg_probackup	

TABLE 11 – Solutions d'absorption de la charge de requête

Nom	Particularité
pgcat	Pooler, Proxy Sharding, Mirroring Multi-threadé
pgBouncer	Pooler
Pgpool-II	Pooler
supavisor	Cloud-native Multi-tenant Pooler
Odyssey	Multi-threadé Pooler et router
pgAgroal	Protocole native Mémoire partagée Validation de connexions



FIGURE 5 – Logos : PGCat, pgBouncer, pgPool-II

TABLE 12 – Solutions de création de file d'attente

Nom	Particularité
pgmq	
PgQ	Lockless queue Based on SQL functions
pgqueueer	Librairie Python

**spqr** Stateless Postgres Query Router  
**vip-manager** Manages a virtual IP based on state kept in etcd or Consul  
**readysset** a MySQL and Postgres wire-compatible caching layer that sits in

front of existing databases to speed up queries and horizontally scale read throughput. Under the hood, ReadySet caches the results of cached select statements and incrementally updates these results over time as the underlying data changes

## Stratégie d’adoption

### Mise en production

TABLE 13 – Solutions de déploiement

Nom	Particularité
spilo	
docker-pgautoupgrade	
TABLE 14 – Solutions d’orchestration	
Nom	Particularité
Patroni	
crunchy-containers	
cloudnative-pg	
postgres-operator	
StackGres	

TABLE 15 – Solutions de développement et d’hébergement cloud

Nom	Particularité
Neon	Stockage à froid Séparation compute-storage Snapshots instantanés
Tembo	Multi-tenant API simplifiée Observabilité intégrée
Supabase	Tems réel, Authentification intégrée Stockage objet (S3-like)
Autobase	Réplication CRDT Gestion conflits automatique Bases distribuées
Omnigres	HTTP/WebSocket natif Logique dans PostgreSQL Mise à jour temps réel



FIGURE 6 – Logos : Tembo, Neon, Supabase, Autobase, Omnigress

TABLE 16 – Framework de développement

Nom	Particularité
Gel	Transactions globales Synchronisation inter-cluster Cohérence forte
pgEdge	Réplication multi-région Bases distribuées Haute disponibilité
Pigsty	Gestion cluster PostgreSQL Monitoring intégré Configuration automatisée
iasql	



FIGURE 7 – Logos : Gel, pgEdge, Pigsty, iaSQL,

## Analyse de compétitive

Analyse comparative : Expliquer pourquoi PostgreSQL résiste face à des concurrents propriétaires (coût, flexibilité) ou open source (MySQL).

Oracle, propriétaire : Coût licence (PostgreSQL = 0) vs Advanced Features Pack (50k) MSSQL, propriétaire : MySQL, Version communautaire GPL-2.0<sup>1</sup> sinon propriétaire : Parallel Query + CTE vs limitations replication

MongoDB, licence "source-accessible" SSPL<sup>2</sup> : Flexibilité schéma (JSONB) + ACID vs AP de MongoDB

— SurrealDB, licence "source-accessible" BSL<sup>3</sup> : All-in-one vs modularité PostgreSQL

— SpacetimeDB, licence "source-accessible" BSL<sup>3</sup> : Blockchain Stored Procedures vs maturité écosystème

— DuckDB, licence open source MIT<sup>4</sup> : OLAP embarqué vs scalabilité Citus

— Elasticsearch, Elastic License<sup>5</sup> ; AGPL<sup>6</sup> ; SSPL<sup>3</sup> : vs ParadeDB

## Conclusion et discussions

## Evolution du document

Date	Changements
2025-03	Rédaction du document

## Glossaire

## Acronymes

## Références

1. JEAN-JACQUES URBAN-GALINDO. *La « Gestion de Configuration » En Ingénierie Ou Ce Que : Tout Responsable Doit Comprendre, Tout Progiciel Devrait Permettre de Maîtriser* [en ligne]. 2008 [visité le 2025-04-03]. Disp. à l'adr.: [https://www.plmlab.fr/public/jjug/a61206%20UGC%20Gest\\_Conf\\_PLN\\_v0\\_97.pdf](https://www.plmlab.fr/public/jjug/a61206%20UGC%20Gest_Conf_PLN_v0_97.pdf).
2. VIDIC, Franc. Knowledge Asset as Competitive Resource. *SocioEconomic Challenges* [en ligne]. 2022, vol. 6, no. 4, p. 8–20 [visité le 2025-01-22]. ISSN 2520-6214, ISSN 2520-6621. Disp. à l'adr. DOI: 10.21272/sec.6(4).8-20.2022.
3. THOUMSIN, Laurence. *Quelles Stratégies Faut-Il Mettre En Place Pour Attirer et Embaucher de Nouveaux Candidats Vers Des Métiers En Pénurie ? Cas Adecco*. 2019. Université catholique de Louvain / Faculté des sciences économiques, sociales, politiques et de communication. Aussi disponible à l'adresse : <http://hdl.handle.net/2078.1/thesis:18580>.
4. LUCIE RIVARD ; MARIE-CHRISTINE ROY. Un Cycle de Rétention Des Connaissances Pour Combattre l'amnésie Organisationnelle. *Télescope*. 2010, t. 16, n° 1, p. 67-81.
5. Zhang, PostgreSQL Wiki [en ligne]. 2021. 12.01 [visité le 2025-02-21]. Disp. à l'adr. <https://www.postgresql.org/docs/12/>