

Member-only story

10 Must-Have PostgreSQL Extensions That Will 10X Your Productivity

3 min read · 1 day ago



Rahul Kumar

Follow

Listen

Share

More



PostgreSQL is one of the most powerful open-source databases, but did you know its **extensions** can supercharge your database capabilities? From **geospatial data** to **real-time analytics**, these add-ons unlock features that normally require expensive proprietary solutions.

After optimizing multiple PostgreSQL databases, I've curated the **10 most impactful extensions** that will save time, cut costs, and open up new possibilities.

PostGIS: Turn PostgreSQL Into a Geographic Database

Use Case: Mapping, location-based queries, geofencing

Why You Need It:

- Store and query **latitude/longitude data** natively.
- Perform complex operations like **distance calculations, spatial joins, and route optimization**.
- Replace specialized tools like MongoDB's geospatial features or Elasticsearch.

Example Query:

```
-- Find all coffee shops within 1km of a given point
SELECT name FROM coffee_shops
WHERE ST_Distance(location, ST_Point(-74.006, 40.7128)) < 1000;
```

How It Helps:

- No need for external **GIS systems**
- Full ACID compliance (unlike NoSQL alternatives)

pg_stat_statements: Find & Fix Slow Queries

Use Case: Performance optimization

Why You Need It:

- Tracks **execution time, calls, and resource usage** of every SQL query.
- Identifies **top 10 slowest queries** in your database.

Example Query:

```
SELECT query, total_time, calls, mean_time
FROM pg_stat_statements
```

```
ORDER BY total_time DESC  
LIMIT 5;
```

How It Helps:

- Pinpoint bottlenecks without guesswork
- Optimize high-traffic queries first

pg_cron: Schedule Jobs Directly in PostgreSQL

Use Case: Automating maintenance, nightly reports

Why You Need It:

- Run SQL queries on a schedule (like cron but inside Postgres).
- No need for external schedulers like Airflow or Celery for simple tasks.

Example:

```
-- Delete old logs every night at 3 AM  
SELECT cron.schedule('0 3 * * *', $$DELETE FROM logs WHERE created_at < NOW() -
```

How It Helps:

- No external dependencies for simple jobs
- Reduce ETL complexity

Citus: Scale PostgreSQL Horizontally

Use Case: High-traffic apps, multi-tenant SaaS

Why You Need It:

- Distributes tables across multiple servers (sharding).
- Works seamlessly with existing PostgreSQL queries.

Example:

```
-- Turn a single-node table into a distributed one
SELECT create_distributed_table('orders', 'user_id');
```

How It Helps:

- Handles 100M+ rows without slowdowns
- Cheaper than proprietary solutions like Amazon Aurora

TimescaleDB: Optimized for Time-Series Data

Use Case: IoT, financial data, monitoring

Why You Need It:

- 10x faster queries on time-based data.
- Automatic partitioning by time (no manual maintenance).

Example:

```
-- Create a hypertable for sensor data
SELECT create_hypertable('sensor_readings', 'time');
```

How It Helps:

- Replace InfluxDB or Prometheus for some use cases
- Built-in compression to save storage

pg_partman: Automatic Table Partitioning

Use Case: Large tables (logs, events)

Why You Need It:

- Auto-splits tables by date, ID, or range.
- Dramatically improves query speed on big datasets.

Example:

```
-- Partition a table by month
SELECT partman.create_parent('public.logs', 'created_at', 'native', 'monthly');
```

How It Helps:

- No more manual `CREATE TABLE` for partitions
- Speeds up `DELETE` operations (drop partitions instead of deleting rows)

pg_trgm: Supercharge Text Search

Use Case: Fuzzy search, autocomplete

Why You Need It:

- Finds “near-matches” (e.g., “Jon Smith” vs. “John Smyth”).
- Works without Elasticsearch for basic search needs.

Example:

```
-- Find similar names (even with typos)
SELECT name FROM users
WHERE similarity(name, 'Michael Jordan') > 0.5;
```

How It Helps:

- No external search index needed
- Faster than `LIKE` for partial matches

hstore: Store Key-Value Pairs in PostgreSQL

Use Case: Flexible schemas, metadata

Why You Need It:

- Add **NoSQL-like** flexibility to PostgreSQL.
- Great for **semi-structured** data (e.g., user preferences).

Example:

```
-- Store and query JSON-like data
UPDATE users SET settings = settings || '{"dark_mode":>"true"}::hstore;
```

How It Helps:

- Avoid schema migrations for minor changes
- Faster than JSONB for simple key-value use cases

uuid-ossp: Generate UUIDs Natively

Use Case: Distributed systems, unique IDs

Why You Need It:

- No more **SERIAL** primary keys (which can conflict in distributed DBs).
- Built-in **UUID** generation (v1, v4, etc.).

Example:

```
-- Create a table with UUID primary key
CREATE TABLE orders (id UUID PRIMARY KEY DEFAULT uuid_generate_v4());
```

How It Helps:

- Prevent ID collisions in microservices
- More secure than auto-incrementing IDs

Foreign Data Wrappers (FDW): Query Other Databases from PostgreSQL

Use Case: Data integration, cross-DB queries

Why You Need It:

- Query MySQL, MongoDB, APIs, and more directly from PostgreSQL.
- No ETL needed for simple joins across systems.

Example:

```
-- Query a remote MySQL table as if it were in PostgreSQL  
SELECT * FROM mysql_mydb.users;
```

How It Helps:

- Unified querying across multiple databases
- Avoid data duplication

Final Thoughts: How to Get Started

- Install extensions with:

```
CREATE EXTENSION IF NOT EXISTS postgis;
```

- Check compatibility with your PostgreSQL version.
- Monitor performance (some extensions add overhead).

Postgresql

Sql

Data Engineering

Data Analysis

Database



Follow

Written by **Rahul Kumar**

61 Followers · 71 Following

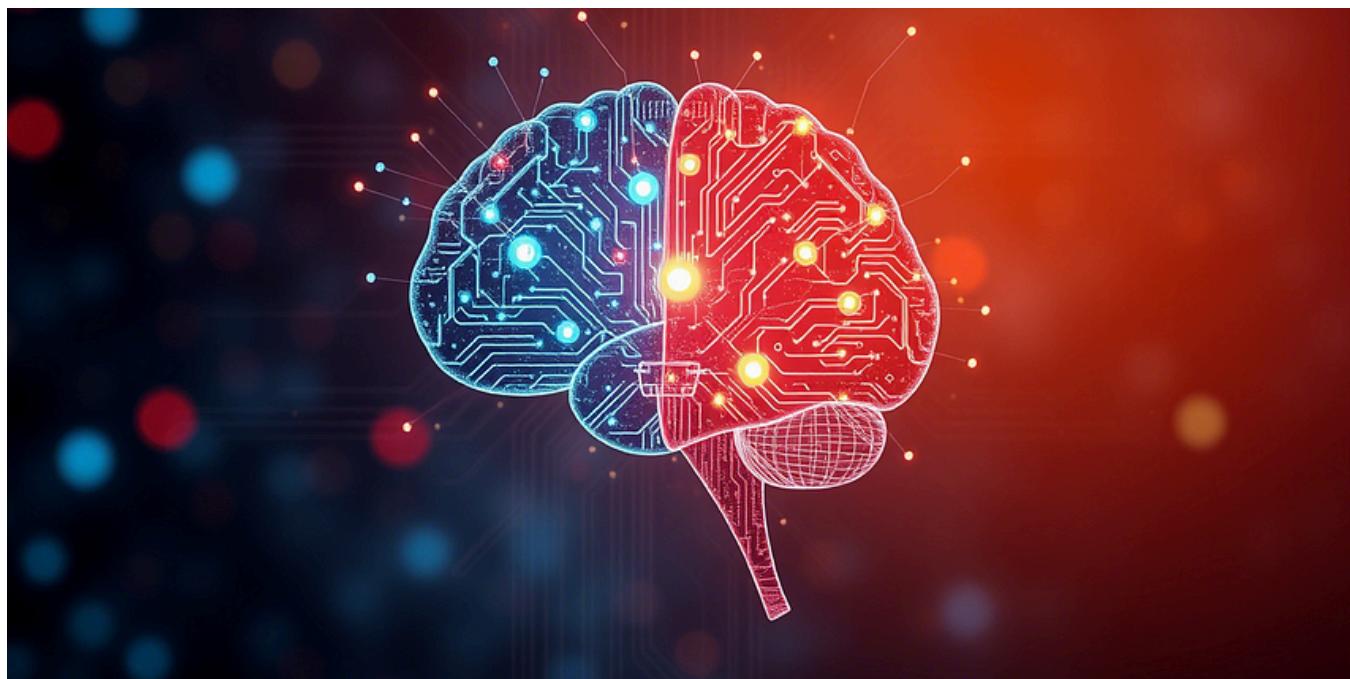
No responses yet



Gvadakte

What are your thoughts?

More from Rahul Kumar

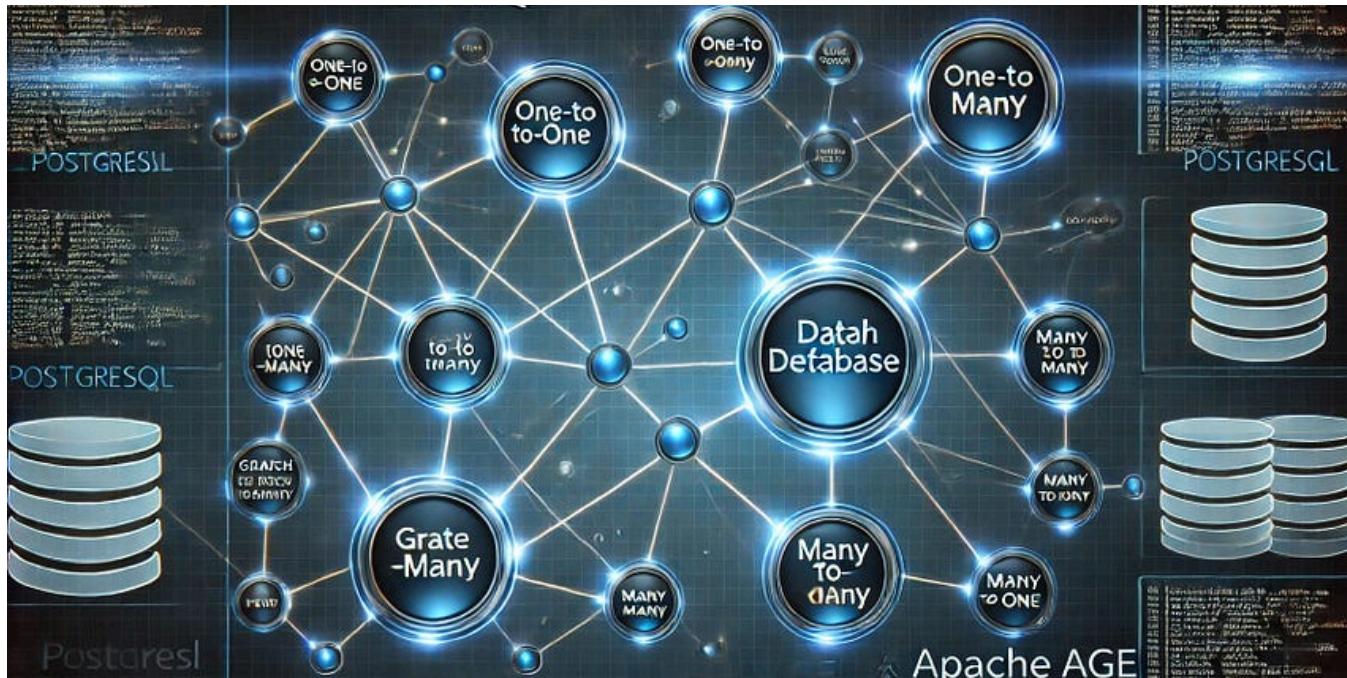


Rahul Kumar

Why Autonomous AI Agents Might Replace Traditional APIs in the Next 5 Years

Exploring how intelligent agents are redefining automation—and why they could outpace microservices and APIs.

Feb 10 58

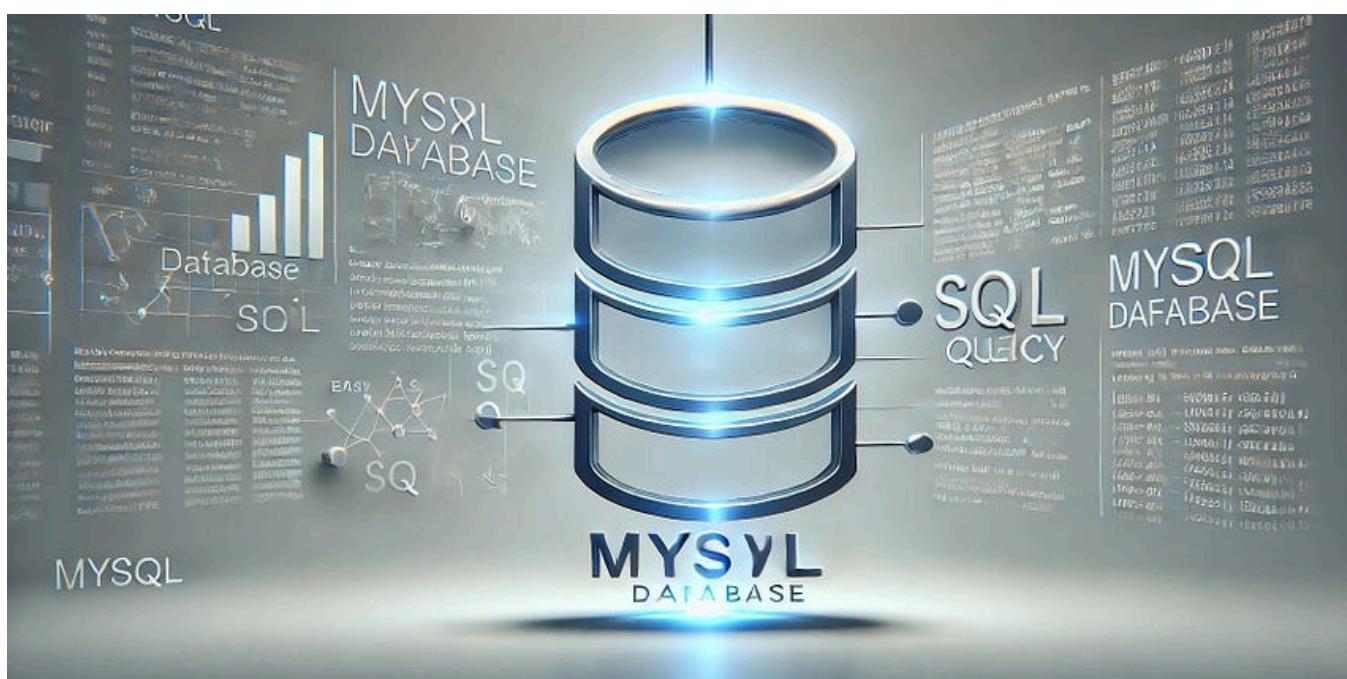


Rahul Kumar

How We Leveraged Apache AGE to Revolutionize Dynamic Data Relationships

In the ever-evolving landscape of enterprise architecture, managing dynamic relationships between customizable data objects has long been a...

Jan 31 6 1



 Rahul Kumar

Stop Writing Nested Queries! How CTEs Made Me a Faster, Smarter, and Wealthier Data Pro

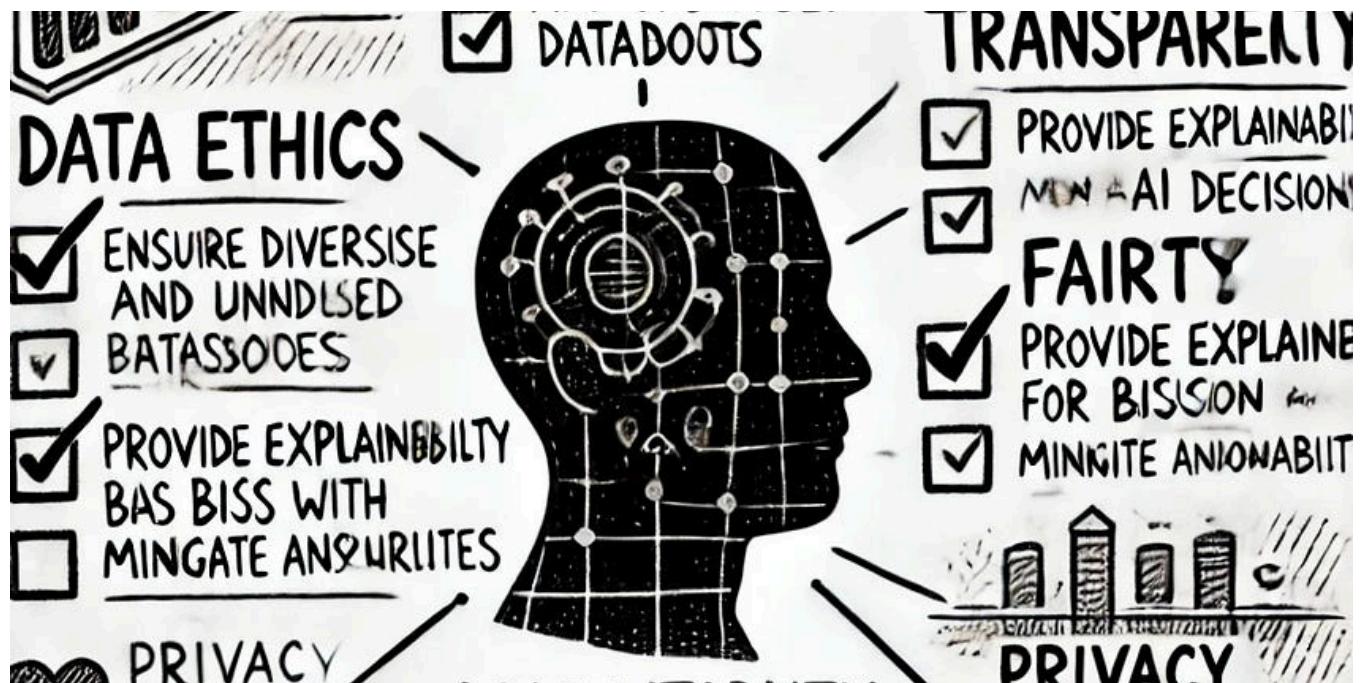
Structured Query Language (SQL) has evolved significantly over the years, and with MySQL 8.0, developers gained access to one of the most...

◆ Feb 2

10



...

 Rahul Kumar

The Dark Side of AI: Unmasking Bias and Ethical Risks

AI is already influencing decisions that shape over 95% of all digital interactions

◆ Mar 4

53

1



...

[See all from Rahul Kumar](#)

Recommended from Medium

[Open in app](#)

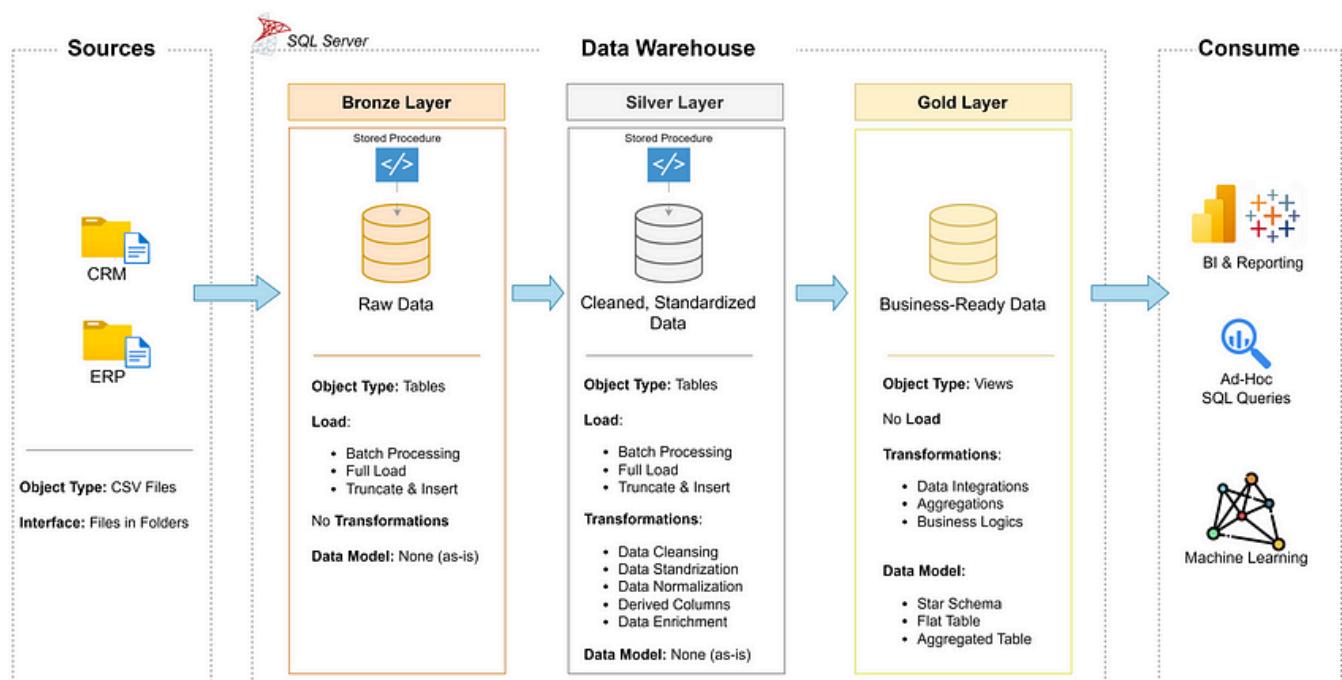


In The Table—Databases and SQL by DbVisualizer

How to Create SQL Temporary Tables: A Practical Guide

SQL temporary tables are useful for handling intermediate data in relational databases. They help optimize queries, stage data, and...

Apr 3



Rihab Feki

Building a modern Data Warehouse from scratch

This project demonstrates a comprehensive data warehousing and analytics solution, from building a data warehouse to generating actionable...

Apr 15 528 15



The screenshot shows the new Google IDE interface. On the left is the Explorer panel with a tree view of a 'MYAPP' project containing files like 'main.dart', 'test', 'web', and 'lib'. The main editor window displays the code for 'main.dart'. To the right of the editor is a preview of a mobile application showing a button and a counter value of 0. Below the preview is a terminal window showing command-line logs related to the app's build and run process.

In Coding Beauty by Tari Ibaba

This new IDE from Google is an absolute game changer

This new IDE from Google is seriously revolutionary.

Mar 12 4.8K 280



The screenshot shows the official PostgreSQL website. At the top, there is a navigation bar with links for 'About', 'Download', 'Documentation', 'Community', 'Developers', 'Support', 'Donate', and 'Your account'. A search bar is located on the right side of the header. The main content area features a large banner with the text 'PostgreSQL: The World's Most Advanced Open Source Relational Database' overlaid on a background image of two people working in a field. Below the banner are two buttons: 'Download →' and 'New to PostgreSQL?'. The footer contains links for 'New to PostgreSQL?' and 'Latest Releases', along with a PostgreSQL logo and a download icon. At the very bottom, there is a small note about PostgreSQL being a powerful open-source database and a link to the latest releases.

Sebastian Petrus

How to Use PostgreSQL MCP Server with Cursor or Claude

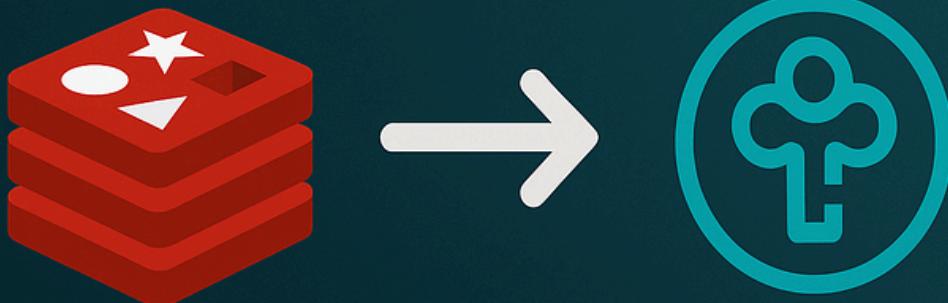
If you love working with PostgreSQL and wish your AI assistant could help manage your database, you're in for something great. The...

Apr 14 ⚡ 47 🎙 4



...

Goodbye Redis, Hello Valkey: A New Chapter in Open Source Databases



howtouselinux

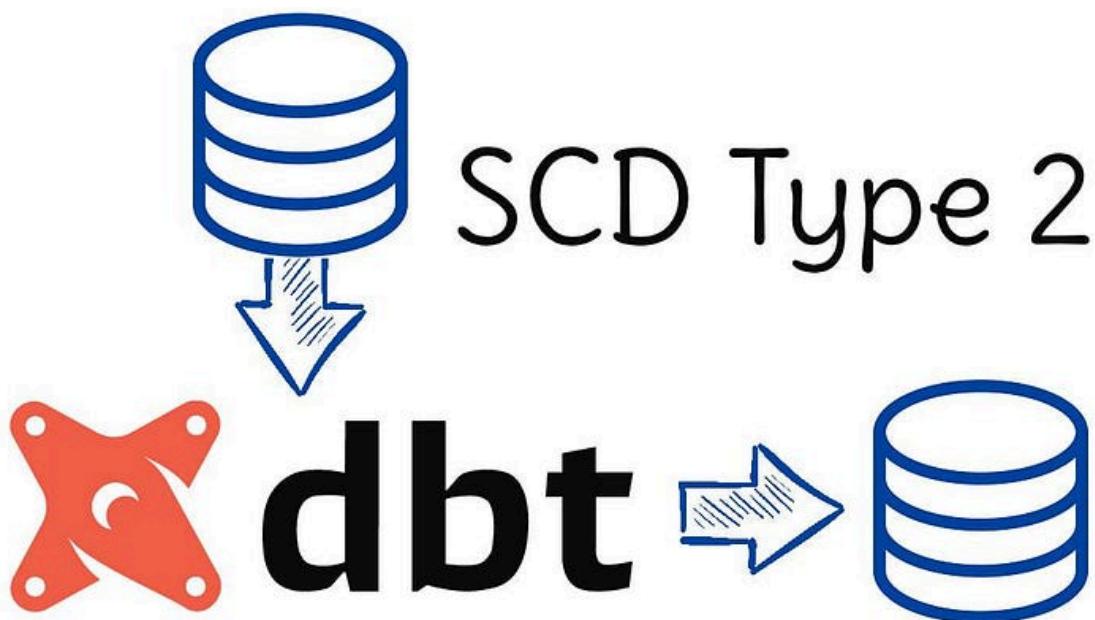
Is This the End for Redis? More Linux Distros Are Saying Goodbye

Another big change is underway in the Linux world—and it's more than just a package swap.

⭐ Apr 20 ⚡ 158



...





Wajahatullah Khan

Using dbt Snapshots to Implement SCD Type 2: A Step-by-Step Guide

Slowly Changing Dimensions (SCDs) are a critical concept in data warehousing that deal with how data changes over time. SCD Type 2 is...

Nov 29, 2024 10



...

See more recommendations