Easy Cross-Database Selects with DuckDB

Josef Machytka

2 min read · Just now

DuckDB was created with simplicity and ease of use in mind. In my previous article, I demonstrated how easily external data can be imported using standard DuckDB commands. In another article, I showcased how to extend DuckDB's functionality with simple Python code, enabling the import of other data formats not natively supported by DuckDB and its extensions.

Today, I want to highlight another powerful use case: cross-database selects. DuckDB currently supports integrations with MySQL, PostgreSQL, and SQLite, allowing users to combine data from these databases within a single query. This allows users to combine data from these databases into a single query within DuckDB, providing enhanced analytical capabilities with minimal effort.

In MySQL, it's possible to select data across databases on the same machine. However, PostgreSQL handles databases differently: even on the same instance, they are isolated from each other. To perform cross-database

selects in PostgreSQL, we must set up foreign data wrappers. Which can be cumbersome, particularly in case of one-time, ad-hoc analyses. The manual work required can quickly become frustrating and counterproductive.

This is precisely where DuckDB excels. It greatly simplifies the entire process of combining data from multiple databases. By attaching databases with specified aliases, we can seamlessly query and combine data across these databases in a single select statement. DuckDB eliminates unnecessary overhead, making cross-database selects a quick and easy process.

For demonstration, I used the classic example of a query joining four different tables: customers, products, orders, and order details. I started three separate Docker containers running PostgreSQL versions 13, 14, and 15, and distributed the tables across these instances. Then, I attached all three databases in DuckDB and executed a combined query without needing to define any additional objects or configurations. The results were immediate and seamless. See the picture below. The same way we could also combine data from PostgreSQL and MySQL.

```
D ATTACH 'host=localhost port=5433 user=postgres password=postgres dbname=test' AS pg13 (TYPE POSTGRES, SCHEMA 'public');
D ATTACH 'host=localhost port=5434 user=postgres password=postgres dbname=postgres' AS pg14 (TYPE POSTGRES, SCHEMA 'public');
D ATTACH 'host=localhost port=5435 user=postgres password=postgres dbname=orders' AS pg15 (TYPE POSTGRES, SCHEMA 'public');
D SELECT
      u.username,
      u.email,
      o.order_date,
      o.total_amount,
      p.product_name,
      od.quantity,
      p.price,
      (od.quantity * p.price) AS total_price
  FROM
      pg13.users u
  JOIN
      pg15.orders o ON u.user_id = o.user_id
  JOIN
      pg15.order_details od ON o.order_id = od.order_id
  JOIN
      pg14.products p ON od.product_id = p.product_id
  ORDER BY
      u.username, o.order_date;
```

username varchar	email varchar	order_date date	total_amount decimal(10,2)	product_name varchar	quantity int32	price decimal(10,2)	total_price decimal(18,2)
Alice	alice@example.com	2024-11-20	150.00	Mouse	2	20.00	40.00
Alice	alice@example.com	2024-11-20	150.00	Keyboard	1	50.00	50.00
Bob	bob@example.com	2024-11-21	250.00	Monitor	1	200.00	200.00
Charlie	charlie@example.com	2024-11-22	300.00	Laptop	1	1000.00	1000.00

Summary

DuckDB sort of revolutionizes many operations with its simplicity and flexibility. It streamlines tasks like importing external data, extending functionality with Python, and performing cross-database selects. By allowing users to simply attach multiple databases and query them simultaneously without too much complexity of different setups, DuckDB offers a straightforward and efficient solution for integrating data across diverse systems for truly efficient data analysis.

Postgresql MySQL Duckdb Etl

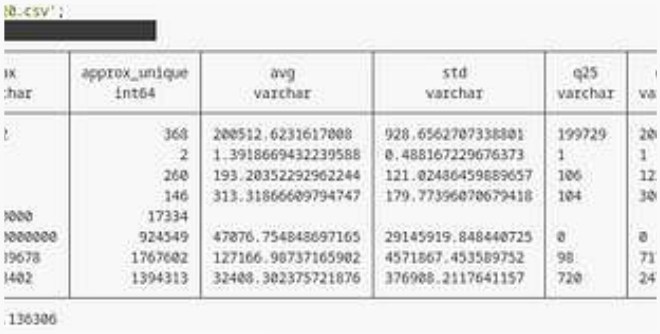
Written by Josef Machytka

9 Followers · 2 Following

Edit profile

I work as Professional Service Consultant - PostgreSQL specialist in NetApp Deutschland GmbH, Open Source Services division.

More from Josef Machytka



	approx_unique int64	avg varchar	std varchar	q25 varchar	
	368	200512.6231617008	928.6562707338801	199729	20
	2	1.3918669432239588	0.488167229676373	1	1
	260	193.20352292962244	121.02486459889657	106	12
	146	313.31866609794747	179.77396070679418	104	30
000	17334				
000000	924549	47076.754848697165	29145919.848440725	0	0
9678	1767602	127166.98737165902	4571867.453589752	98	71
402	1394313	32408.302375721876	376908.2117641157	720	24

Josef Machytka

How DuckDB handles data not fitting into memory?

In my previous article about DuckDB I described how to use this database as an...

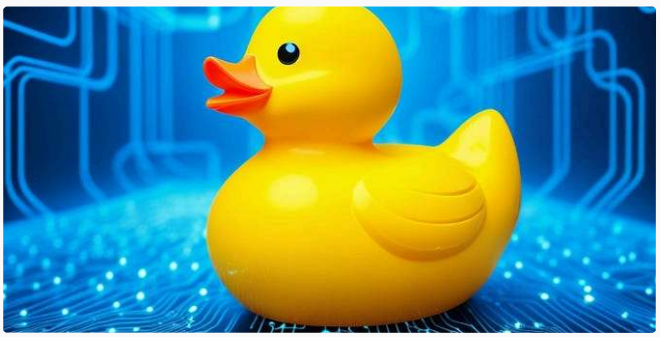
Nov 13 1

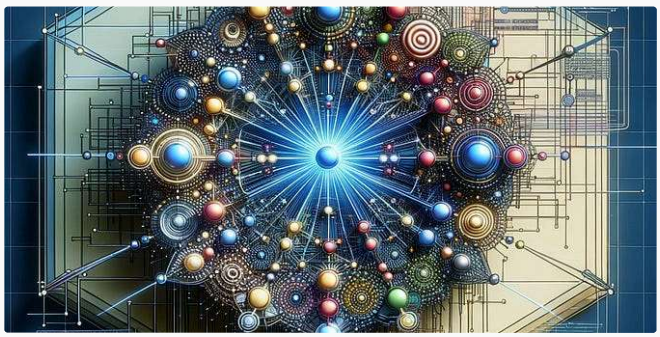
Josef Machytka

Using DuckDB as an Intelligent ETL tool for PostgreSQL

There is a lot of hype around DuckDB these days. At one PostgreSQL conference, I even...

Nov 2





Josef Machytka

Extending DuckDB ETL Capabilities with Python

DuckDB has recently become my go-to solution for small ETL tasks. It is an...

2d ago 6

Josef Machytka

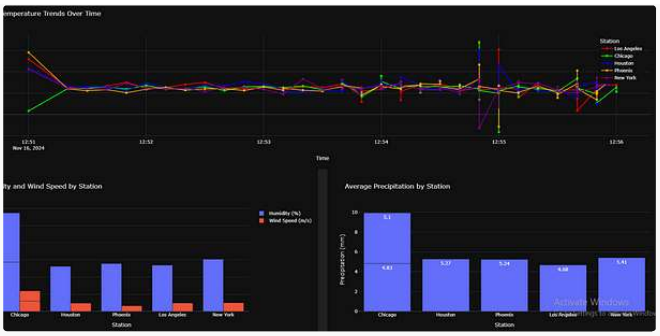
AI Hallucinations are caused by Quantum Pigeons Nesting in...

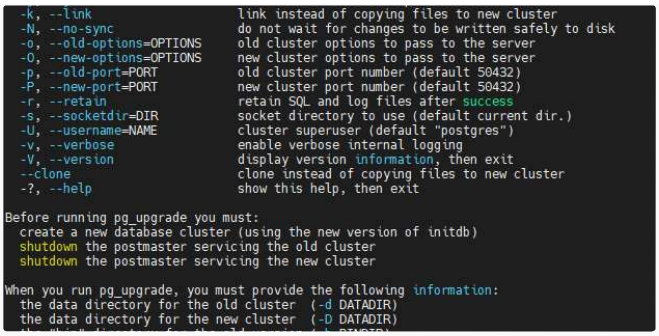
This is not a new discovery in quantum physics—it is a playful deliberate...

Nov 5

See all from Josef Machytka

Recommended from Medium





Wissem Hammoudi

Building a Scalable Data Pipeline: A Step-by-Step Guide with Kafka,...

Introduction

Nov 16 3

Sheikh Wasiu Al Hasib

Upgrading PostgreSQL major version using `pg_upgrade`

Upgrading PostgreSQL from version 14 to 15 can be done using `pg_upgrade`, which is a...

Jul 25 1

Lists

Staff picks

775 stories · 1465 saves

Stories to Help You Level-Up at Work

19 stories · 877 saves

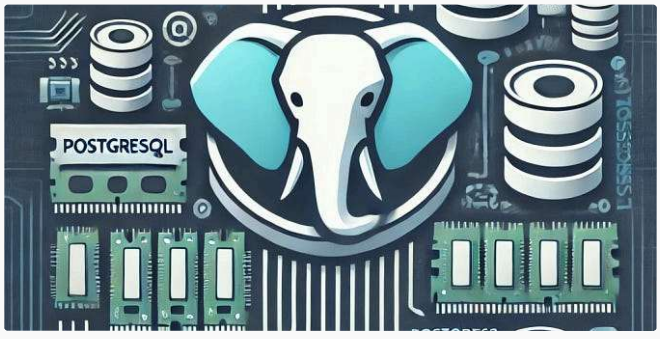
Self-Improvement 101

20 stories · 3086 saves

Productivity 101

20 stories · 2596 saves

Hugo Lu



Hüseyin Demir

What Snowflake's Acquisition of Datavolo means for the Data...

Cloudera, Hortonworks, Unstructured Data, and of course—AI

5d ago 67 3

Memory Matters in PostgreSQL : Configuring max_connections an...

Hello everyone! In this blog post, I'll discuss the relationship between the work_mem (an...

Nov 9 6

In Python in Plain English by Satyam Sahu

How to Build a Data Pipeline for API Integration Using Python and...

A hands-on approach to fetching, storing, and analyzing data from APIs

Nov 18 53

In Towards Data Engineering by Burak Uğur

Create Data Lakehouse Using Spark+Iceberg+Nessie+Dremio

Hi everyone, in this article I will talk about the concept of data lakehouse and develop...

Sep 19 75

See more recommendations