Towards Dev · Follow publication

How to combine Pgbackrest and Streaming Replication using Postgresql-16

5 min read · Mar 1, 2025

MynotesoracleDBA Follow

Listen Share More

In this blog we are going to discuss about Combining pgBackRest and Streaming Replication

Pre-req:

- Backup server — Pgbackrest–172.18.0.30
- Primary server — Postgresql-16–172.18.0.31
- Standby server — Postgresql-16–172.18.0.32

Source:Google

In Backup server:

S-1: Install pgBackRest

Install pgBackRest on both the primary and standby servers:

```
dnf install -y https://download.postgresql.org/pub/repos/yum/reporpms/EL-8-x86_
dnf -qy module disable postgresql
dnf install -y pgbackrest
```

Pgbackrest Installation You can follow Step-1 to Step-5 from this blog

S-2: Configure Replication

Install the postgresql-16 and enable these below parameters to configure streaming replication

To configure streaming replication betweeen primary (172.18.0.31) to standby (172.18.0.32) you can refer this blog How to configure Streaming replication

S-3: Verify the replication

We can verify the replication status using below

```
postgres=# select * from pg_stat_replication;
-[ RECORD 1 ]----+------------------------------
pid              | 389822
usesysid         | 10
usename          | postgres
application_name | walreceiver
client_addr      | 172.18.0.32
client_hostname  |
client_port      | 52810
backend_start    | 2025-03-01 14:48:55.807108+00
backend_xmin     |
state            | streaming
sent_lsn         | 0/E000148
write_lsn        | 0/E000148
flush_lsn        | 0/E000148
replay_lsn       | 0/E000148
write_lag        |
flush_lag        |
replay_lag       |
sync_priority    | 0
sync_state       | async
reply_time       | 2025-03-01 16:34:40.901623+00
```

S-4: To Configure Pgbackrest

On Backup server side:

```
[postgres@pg01 pgbackrest]$ cat /etc/pgbackrest.conf

[global]
repo1-path=/var/lib/pgbackrest
repo1-retention-full=1
start-fast=y
log-level-console=info
log-level-file=debug
process-max=2
archive-timeout=1h
compress-level=3
backup-standby=y

[mycluster]
pg1-host=172.18.0.31
pg1-host-user=postgres
pg1-port=5432
pg1-path=/var/lib/pgsql/16/data
pg2-host=172.18.0.32
```

```
pg2-host-user=postgres
pg2-port=5432
pg2-path=/var/lib/pgsql/16/data
```

On Primary side:

```
postgres@pg02 ~]$ sudo cat /etc/pgbackrest.conf


[global]
repo1-host=172.18.0.30
repo1-host-user=postgres
repo1-path=/var/lib/pgbackrest
repo1-retention-full=1
start-fast=y
log-level-console=info
log-level-file=debug
compress-type=zst
compress-level=3


[mycluster]
pg1-path=/var/lib/pgsql/16/data
pg1-port=5432
```

On Standby side:

```
[postgres@pg03 log]$ sudo cat /etc/pgbackrest.conf

[global]
repo1-host=172.18.0.30
repo1-host-user=postgres
repo1-path=/var/lib/pgbackrest
repo1-retention-full=1
start-fast=y
log-level-console=info
log-level-file=debug
compress-type=zst
compress-level=3

[mycluster]
```

```
pg1-path=/var/lib/pgsql/16/data
pg1-port=5432
```

S-5: To Configure archive command :

Edit the postgresql.conf file on primary side modify these parameters and save the file

```
archive_mode=on
archive_command='pgbackrest --stanza=mycluster archive-push %p'
```

Restart the cluster

```
sudo systemctl restart postgresql-16
```

S-6: Set up Password less authentication

1. On the backup server, generate SSH keys:

```
sudo -u postgres ssh-keygen -t rsa
```

2.Copy the public key to both primary and standby servers:

```
sudo -u postgres ssh-copy-id postgres@172.18.0.31
sudo -u postgres ssh-copy-id postgres@172.18.0.32
```

3.Test the connections:

```
sudo -u postgres ssh postgres@172.18.0.31
sudo -u postgres ssh postgres@172.18.0.32
```

Note: You have provided necessary permission on these below files

```
By default, the packages create the following directories owned by postgres:

sudo chown -R postgres: /var/lib/pgbackrest
sudo chown -R postgres: /var/log/pgbackrest
sudo chown -R postgres: /var/spool/pgbackrest
```

S-6: Create Stanza

On Your Backup server you can create a stanza using below

```
[postgres@pg01 pgbackrest]$ pgbackrest --stanza=mycluster stanza-create
```

Verify the check command on primary and standby

```
postgres@pg02 ~]$ pgbackrest --stanza=mycluster --log-level-console=detail check
```

```
[postgres@pg03 ~]$  pgbackrest --stanza=mycluster --log-level-console=detail ch
```

Make sure not seeing any errors over here

S-7: Perform a Full Backup

```
[postgres@pg01 ~]$ pgbackrest --stanza=mycluster --log-level-console=detail bac
```

Incase if your archiving is not happening/not configure properly you can see the error messages

verify the backup

```
[postgres@pg02 ~]$ pgbackrest --stanza=mycluster info
stanza: mycluster
    status: ok
    cipher: none

    db (current)
        wal archive min/max (16): 00000001000000000000000D/00000001000000000000

        full backup: 20250301-145458F
            timestamp start/stop: 2025-03-01 14:54:58+00 / 2025-03-01 14:55:03+
            wal start/stop: 00000001000000000000000D / 00000001000000000000000D
            database size: 22.6MB, database backup size: 22.6MB
            repo1: backup set size: 3.2MB, backup size: 3.2MB
```

S-8: Perform backup from standby

pgBackRest can perform backups on a standby server instead of the **primary**. Both the **primary** and **secondary** databases configuration are required, even if the majority of the files will be copied from the **secondary** to reduce load on the **primary**.

Remark: To do so, you need to setup a trusted SSH communication between the hosts.

SSH to backup server perform backup from standby,alternatively execute the backup command from backupserver because The backup server (172.18.0.30) acts as a central point for managing backups .It stores all backup files and manages the backup process for both primary and standby servers.

```
pgbackrest --stanza=mycluster --type=incr --backup-standby backup
2025-03-01 17:14:12.507 P00   INFO: backup command begin 2.54.2: --archive-time
2025-03-01 17:14:13.390 P00   INFO: last backup label = 20250301-145458F, versi
2025-03-01 17:14:13.390 P00   INFO: execute non-exclusive backup start: backup
2025-03-01 17:14:13.444 P00   INFO: backup start archive = 000000010000000000000
2025-03-01 17:14:13.444 P00   INFO: wait for replay on the standby to reach 0/1
2025-03-01 17:14:13.562 P00   INFO: replay on the standby reached 0/10000028
2025-03-01 17:14:13.562 P00   INFO: check archive for prior segment 00000001000
2025-03-01 17:14:15.995 P00   INFO: execute non-exclusive backup stop and wait
2025-03-01 17:14:16.094 P00   INFO: backup stop archive = 0000000100000000000000
2025-03-01 17:14:16.104 P00   INFO: check archive for segment(s) 00000001000000
2025-03-01 17:14:17.317 P00   INFO: new backup label = 20250301-145458F_2025030
2025-03-01 17:14:17.344 P00   INFO: incr backup size = 611.3KB, file total = 96
2025-03-01 17:14:17.345 P00   INFO: backup command end: completed successfully
2025-03-01 17:14:17.345 P00   INFO: expire command begin 2.54.2: --exec-id=4869
2025-03-01 17:14:17.347 P00   INFO: repo1: 16-1 no archive to remove
2025-03-01 17:14:17.347 P00   INFO: expire command end: completed successfully
```

Verify the backup:

```
[postgres@pg01 pgbackrest]$ pgbackrest --stanza=mycluster info
stanza: mycluster
    status: ok
    cipher: none

    db (current)
        wal archive min/max (16): 00000001000000000000000D/000000010000000000000

            full backup: 20250301-145458F
                timestamp start/stop: 2025-03-01 14:54:58+00 / 2025-03-01 14:55:03+
                wal start/stop: 00000001000000000000000D / 00000001000000000000000D
                database size: 22.6MB, database backup size: 22.6MB
                repo1: backup set size: 3.2MB, backup size: 3.2MB

            incr backup: 20250301-145458F_20250301-171413I
                timestamp start/stop: 2025-03-01 17:14:13+00 / 2025-03-01 17:14:16+
                wal start/stop: 000000010000000000000010 / 000000010000000000000010
                database size: 22.6MB, database backup size: 611.3KB
                repo1: backup set size: 3.2MB, backup size: 27.0KB
                backup reference total: 1 full
[postgres@pg01 pgbackrest]$
```

This command will initiate a full backup from the standby server. The `--backup-standby` option tells pgBackRest to use the standby for the backup.

Pgbackrest Backup Standby Backup And Restore Postgresql

Follow

Published in Towards Dev

7.3K Followers · Last published 20 hours ago

A publication for sharing projects, ideas, codes, and new theories.

Follow

Written by MynotesoracleDBA

313 Followers · 14 Following

As a DBA we are going to write and discuss multiple database administration concepts. “Nothing Grows In Comfort Zone”.

No responses yet

Gvadakte

What are your thoughts?

More from MynotesoracleDBA and Towards Dev

In Nerd For Tech by MynotesoracleDBA

How a SQL Statement is processed in Oracle?| Interview Q& A

Here we are going to understand the SQL statement processing steps in oracle

Dec 7, 2021 3

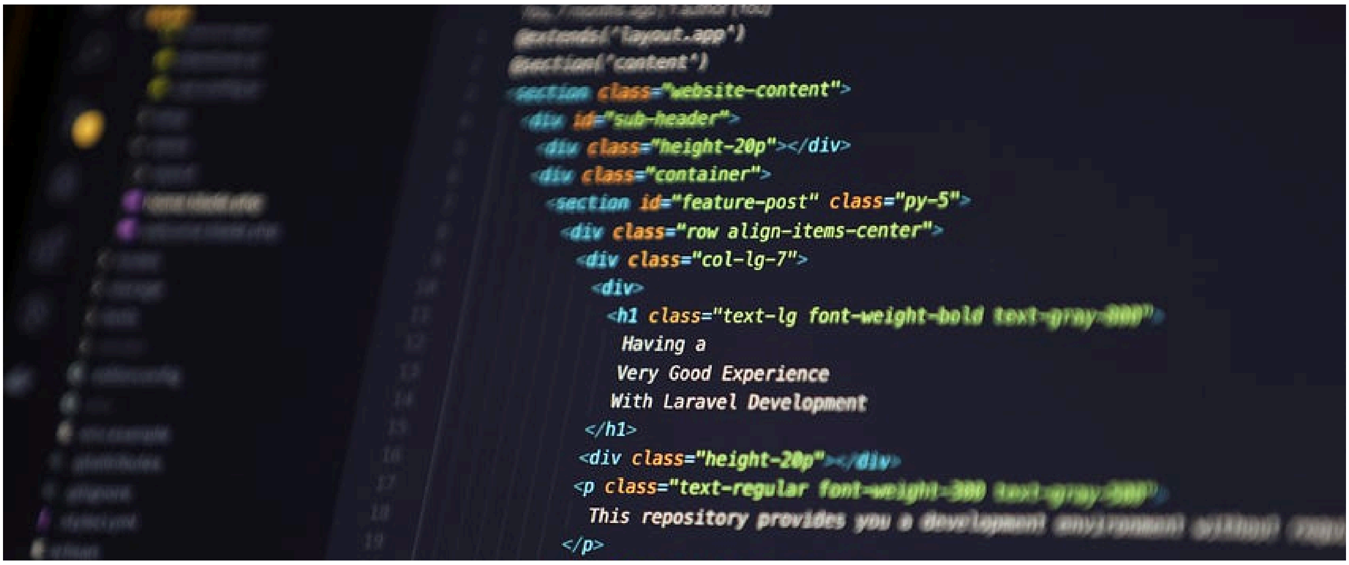
10 Costly HttpClient Mistakes in
C#.NET (And How to Fix Them Like a
Pro)

In Towards Dev by AshokReddy

10 Costly HttpClient Mistakes in C#.NET (And How to Fix Them Like a Pro)

Introduction:

✦ Apr 12 151 6



```
@extends('layout.app')
@section('content')
<section class="website-content">
  <div id="sub-header">
  <div class="height-20p"></div>
  <div class="container">
   <section id="feature-post" class="py-5">
    <div class="row align-items-center">
     <div class="col-lg-7">
      <div>
       <h1 class="text-lg font-weight-bold text-gray-800">
        Having a
        Very Good Experience
        With Laravel Development
       </h1>
      <div class="height-20p"></div>
      <p class="text-regular font-weight-300 text-gray-500">
       This repository provides you a development environment without
      </p>
```

Open in app ↗

Medium Search

This Is How I Scrape 99% of Websites (Step-by-Step Guide)

“Without data, you’re just another person with an opinion.” — W. Edwards Deming

✦ Jan 15 703 11

In Nerd For Tech by MynotesoracleDBA

What Is A PGA In Oracle?

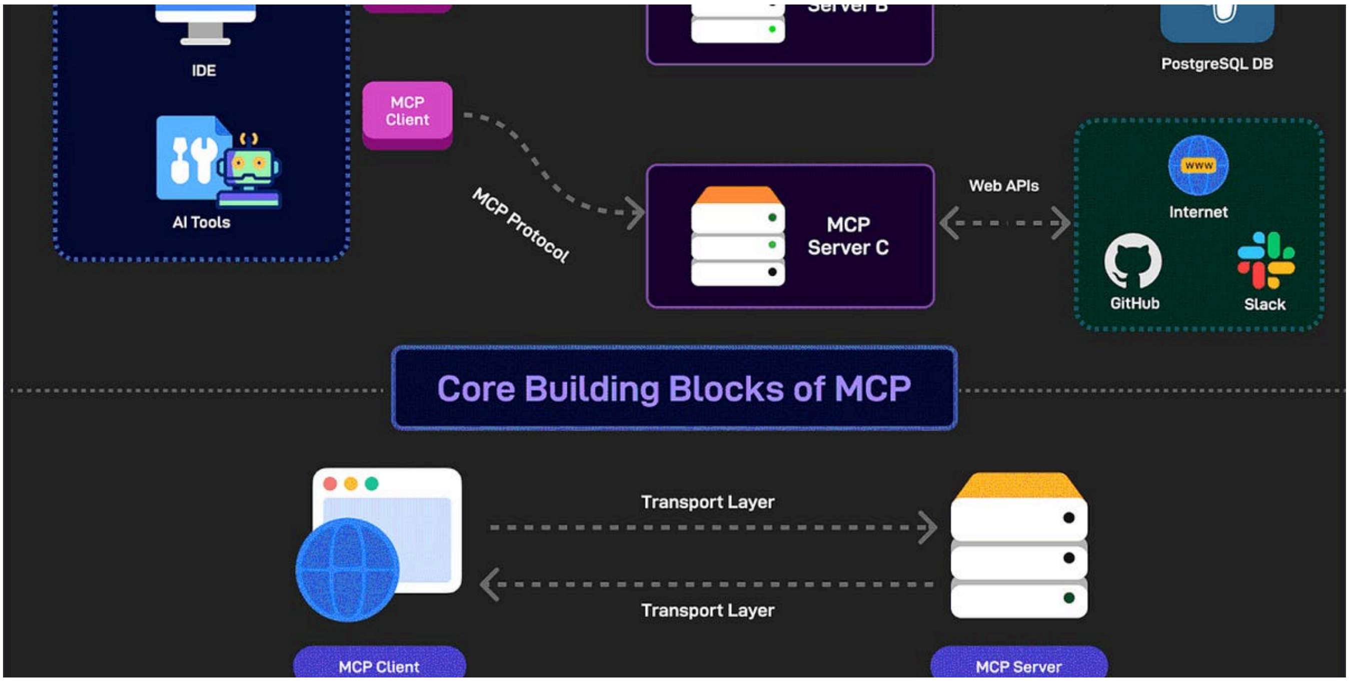
Here we are going to understand the components of PGA

Feb 6, 2022 5

See all from MynotesoracleDBA

See all from Towards Dev

Recommended from Medium



In Towards AGI by Gary Svenson

How to Setup and Use PostgreSQL MCP Server

Introduction to PostgreSQL MCP Server

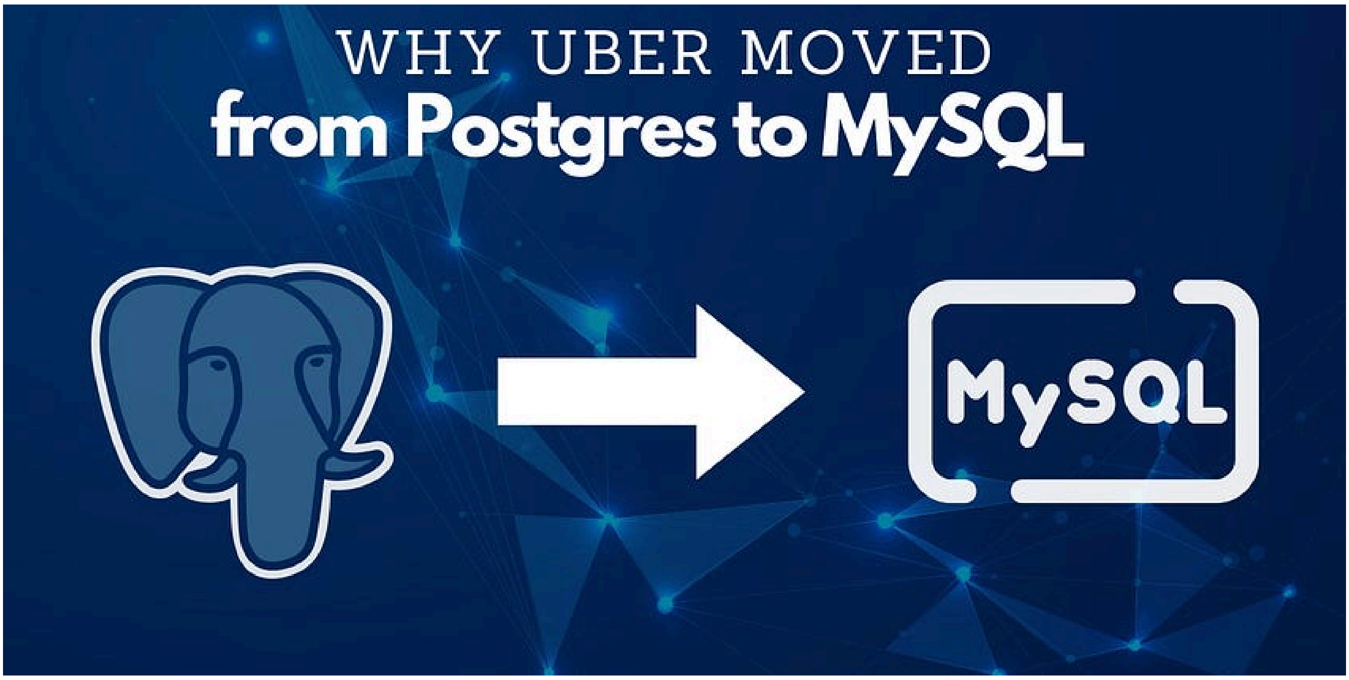
Apr 8 85

Analyst Uttam

This SQL Trick Cut My Query Time by 80%

🚪 The Problem That Wasted My Hours (And Sanity)

✦ Apr 22 👏 1.1K 💬 42

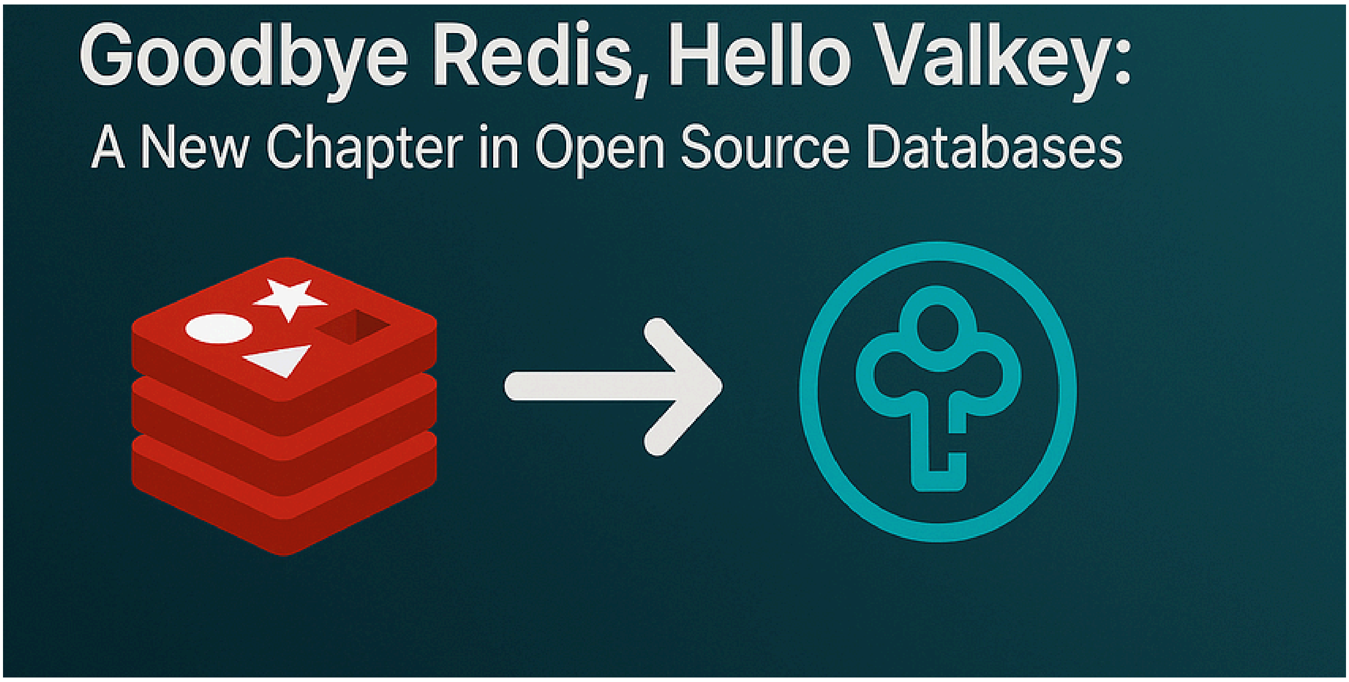


In Databases by Sergey Egorenkov

Why Uber Moved from Postgres to MySQL

How PostgreSQL's architecture clashed with Uber's scale—and why MySQL offered a better path forward

Mar 29 1.2K 29

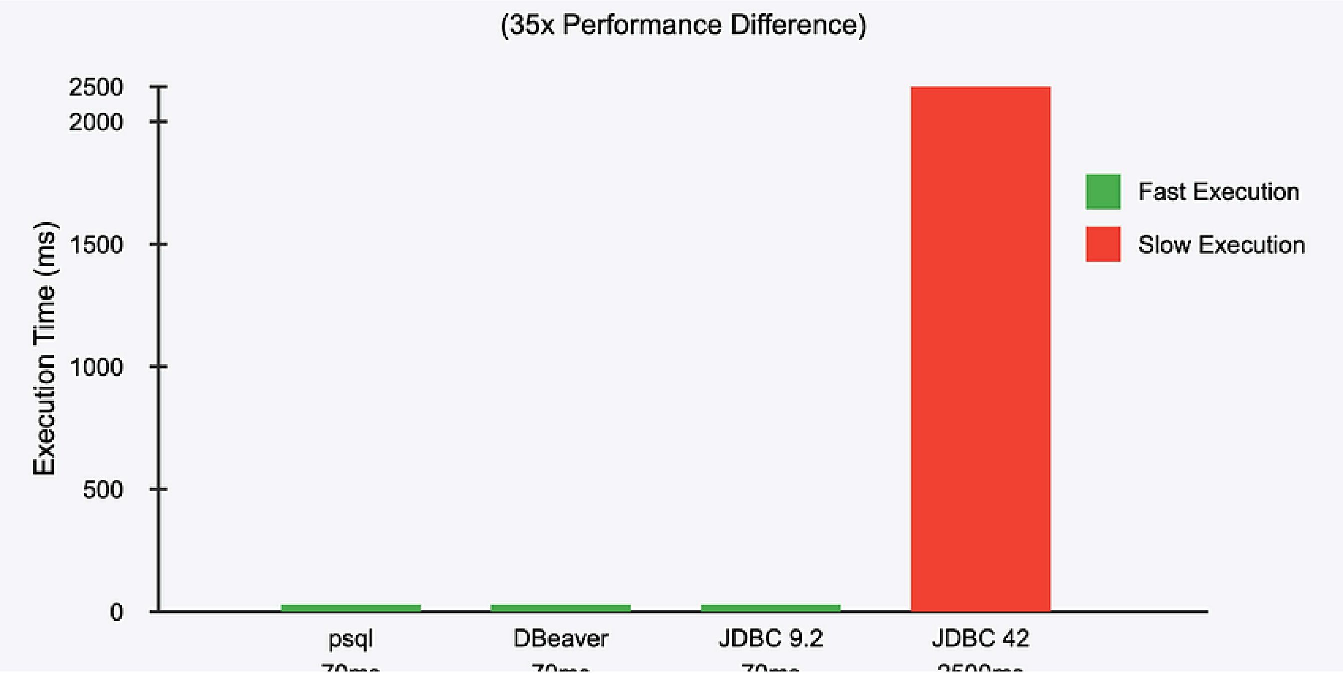

Goodbye Redis, Hello Valkey:
A New Chapter in Open Source Databases

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 20 378 1



Shailesh Kumar Mishra

Solving PostgreSQL Performance Issues with JDBC Drivers: The Surprising Impact of Prepared...

1. Introduction

Mar 16 26 1

Sruthi Ganesh

10 Best Practices to Optimize Your PostgreSQL Database (with Examples)

PostgreSQL is a robust and feature-rich database, but maximizing its performance requires intentional design and ongoing management. Below...

Nov 20, 2024 9

See more recommendations