

# **CERN PGDay 2025**

Friday 17 January 2025 - Friday 17 January 2025

CERN

## **Book of Abstracts**



# Contents

Architecture for Seamless PostgreSQL Upgrades . . . . .	1
Closing . . . . .	1
Anatomy of Table-Level Locks in PostgreSQL . . . . .	1
From Queries to Pints: Building a Beer Recommendation System with pgvector . . . . .	2
Maximising Microservice Databases with Kubernetes, Postgres, and CloudNativePG . . . . .	2
Access control below table level (This is a reserve talk: presented instead of someone else's talk ONLY in case of forfeit) . . . . .	3
Optimizing time series data storage for CERN industrial control systems using TimescaleDB and PostgreSQL . . . . .	3
Harnessing Postgres and HPC for Petabyte-Scale Variable Star Classification in Astronomy . . . . .	4
Opening . . . . .	4
CERN introductory keynote: The role of PostgreSQL at CERN . . . . .	4



15

## Architecture for Seamless PostgreSQL Upgrades

**Author:** Adrien Obernesser<sup>1</sup>

<sup>1</sup> *dbi services*

**Corresponding Author:** adrien.obernesser@dbi-services.com

Upgrading from PostgreSQL 11 to 16 can be an easy process, such as a dump restore, but when instance numbers and sizes increase, this can push organizations to maintain older PostgreSQL versions longer than expected. Avoiding technological debt is crucial for maintaining a robust and scalable database architecture. This session will present a real-world scenario on how to design an architecture that allows for easy migration between PostgreSQL versions. By using load balancers like HAProxy for connection management, PgBouncer for connection pooling, and automated logical replication for testing and migration, we can achieve seamless upgrades and minimize downtime. Additionally, we will explore how the new `pg_create_subscriber` feature in PostgreSQL 17 will further simplify this process, making future migrations even more efficient. The ultimate goal is to upgrade to new releases as soon as possible and allow developers to run all the necessary tests easily, simplifying the rollback process.

29

## Closing

**Author:** Tobias Bussmann<sup>1</sup>

**Co-author:** Maurizio De Giorgi<sup>2</sup>

<sup>1</sup> *Swiss PostgreSQL Users Group*

<sup>2</sup> *CERN*

**Corresponding Authors:** maurizio.degiorgi@cern.ch, t.bussmann@gmx.net

Closing notes with sponsors review and special thanks

32

## Anatomy of Table-Level Locks in PostgreSQL

**Author:** Gulcin Yildirim Jelinek<sup>None</sup>

**Corresponding Author:** gulcin@xata.io

### Short abstract

Understanding table-level locks in Postgres is a quite useful skill as almost all DDL operations require acquiring one of the different types of table-level locks on the object being manipulated. If not managed well, schema changes can result in downtime. In this talk we will explain fundamentals of table-level locking, covering how different types of locks are applied and queued during schema changes. Attendees will learn how to identify and manage lock conflicts to minimize downtime, avoid deadlocks, and maintain smooth database operations, even during high-concurrency schema changes.

### Long Abstract

In PostgreSQL, managing schema changes without downtime can be a challenging task. Table-level locks, which control access during Data Definition Language (DDL) operations like ALTER or DROP

TABLE, can result in unintended application slowdowns or even service interruptions when not fully understood. This talk will provide a comprehensive dive into table-level locking and lock queueing in PostgreSQL, helping attendees gain the insights they need to perform efficient schema changes.

We'll start by explaining the various types of table-level locks in PostgreSQL such as Access Share, Exclusive, and Access Exclusive and how they are automatically managed during common DDL operations. Then, we'll break down lock queueing: how PostgreSQL organizes lock requests, what happens when transactions wait for locks, and how deadlocks can arise in complex environments.

Next, we'll focus on practical approaches to managing table-level locks for near-zero downtime. Attendees will learn techniques to minimize locking impact, including understanding lock conflicts, using online schema migration patterns, and identifying lock-heavy queries. We'll introduce open-source tools like pgsroll, which utilizes the expand/contract pattern to make schema changes in small, lock-free steps.

By the end of this session, attendees will be equipped with practical strategies and knowledge to control lock behavior during schema changes, ensuring data integrity and reducing operational disruptions. This talk will provide the tools needed to manage PostgreSQL schema changes with confidence and minimal impact on production environments.

41

## From Queries to Pints: Building a Beer Recommendation System with pgvector

**Author:** Andrzej Nowicki<sup>1</sup>

<sup>1</sup> CERN

**Corresponding Author:** andrzej.nowicki@cern.ch

Discover how easily you can create a recommendation system from scratch using modern AI and database technologies.

In this session, we'll build a beer recommendation system using advanced language models and PostgreSQL's pgvector extension. By leveraging the capabilities of pgvector, we can seamlessly store high-dimensional embeddings generated from beer descriptions and perform similarity search with user preferences.

Whether you're a seasoned database administrator or just starting to explore the potential of AI, this presentation will equip you with practical insights and hands-on techniques to integrate machine learning into your database workflows. Join to learn how to turn complex data into intuitive recommendations! No prior ML knowledge required.

43

## Maximising Microservice Databases with Kubernetes, Postgres, and CloudNativePG

**Authors:** Gabriele Bartolini<sup>None</sup>; Leonardo Cecchi<sup>None</sup>

**Corresponding Authors:** gabriele.bartolini@enterprisedb.com, leonardo.cecchi@enterprisedb.com

As a developer, you may follow continuous delivery practices yet treat your database as an external component, introducing risks with changes and migrations. It's time to change that by embracing microservice databases within your organisation.

This talk explores how PostgreSQL and Kubernetes, through CloudNativePG, can help you establish a seamless, open-source, vendor-neutral microservice database environment. We'll discuss how adopting technical DevOps capabilities enables multidisciplinary development teams to create microservice-based applications that integrate smoothly with their databases, owning the entire flow to the end user.

This approach eliminates dependencies that translate into bottlenecks, reduces cognitive load, improves quality, and enhances agility. Additionally, PostgreSQL's ACID transactional DDL simplifies database change management, allowing for complete rollbacks in case of errors.

47

## **Access control below table level (This is a reserve talk: presented instead of someone else's talk ONLY in case of forfeit)**

**Author:** Alastair Turner<sup>None</sup>

**Corresponding Author:** minion@cjt.london

No, not under the table - for things within a table, like columns and rows. Fine grained access controls let administrators manage access to specific pieces of data. Granting a user permission to read a piece of data doesn't have to mean access to all the details either, values can be masked, rounded, classified into buckets, ... As regulation makes data privacy controls more important and the tools for exploring and exploiting data become more diverse, enforcing these controls in the database is an appealing option for ensuring that the enforcement is consistent.

Postgres column-level permissions, row-level security, security labels, and the pg\_anonymize extension offer advanced mechanisms for controlling access to data. Join me for an overview of how you can use them to control exactly who sees what.

54

## **Optimizing time series data storage for CERN industrial control systems using TimescaleDB and PostgreSQL**

**Authors:** Rafal Kulaga<sup>1</sup>; Antonin Kveton<sup>1</sup>; Martin Zemko<sup>1</sup>

<sup>1</sup> CERN

**Corresponding Authors:** antonin.kveton@cern.ch, rafal.lukasz.kulaga@cern.ch, martin.zemko@cern.ch

CERN is known for producing, processing, and storing vast amounts of data, most of which consists of physics-related information — snapshots of particle collisions recorded by detectors in experiments. Large Detector Control Systems at CERN enable the controlled operation of complex research equipment and generate “conditions” data. This data describes the time evolution of parameters such as voltages, pressures, and temperatures, which are essential for physics data analysis and ensuring reproducibility.

These control systems include over 800 mission-critical SCADA systems built on SIMATIC WinCC Open Architecture software, developed by ETM (Siemens). Together, they produce hundreds of gigabytes of time-series data daily. Storing and querying this data efficiently is the responsibility of the NextGen Archiver module of WinCC OA, a solution developed collaboratively by CERN and Siemens through the CERN openlab project. The ever-growing data volumes and throughput, combined with the need for fast, responsive user interfaces for data visualization, have posed significant challenges for the project.

In this presentation, we will share the lessons learned during the development of the NextGen Archiver. The NextGen Archiver supports multiple database technologies through pluggable back-ends, offering PostgreSQL and TimescaleDB as alternatives to Oracle Database, which has been used for WinCC OA archiving for over 15 years.

We will walk you through the processes and tools used to evaluate the performance and scalability of various database technologies and schemas considered during the project. Additionally, we will examine the impact of TimescaleDB features, such as compression and continuous aggregates, on improving query performance and reducing storage requirements.

55

## Harnessing Postgres and HPC for Petabyte-Scale Variable Star Classification in Astronomy

**Authors:** Daniel Krefl<sup>1</sup>; Krzysztof Nienartowicz<sup>1</sup>

<sup>1</sup> *Sednai*

In this talk, we will showcase our use of a Massively Parallel Postgres Database as a petabyte-scale astronomical data store, powering variable star analysis for the ESA Gaia space telescope. Our presentation will also highlight two Postgres extensions developed in-house: a new Postgres-to-Java interface and a GPU-accelerated vector index. Both extensions are implemented as Postgres background workers and improve on existing solutions in the HPC context.

59

## Opening

**Author:** Tobias Bussmann<sup>1</sup>

**Co-author:** Maurizio De Giorgi<sup>2</sup>

<sup>1</sup> *Swiss PostgreSQL Users Group*

<sup>2</sup> *CERN*

**Corresponding Authors:** maurizio.degiorgi@cern.ch, t.bussmann@gmx.net

Welcome from the organizing team, logistics info and notices, sponsors presentation and special thanks

60

## CERN introductory keynote: The role of PostgreSQL at CERN

**Author:** Miroslav Potocky<sup>1</sup>

<sup>1</sup> *CERN*

**Corresponding Author:** miroslav.potocky@cern.ch



CERN, the European Organization for Nuclear Research, is at the forefront of scientific exploration, uncovering the fundamental nature of the universe. With groundbreaking experiments from worldwide collaborations conducted at the Large Hadron Collider (LHC), the world's largest and most powerful particle accelerator. The LHC hosts four main experiments: ATLAS and CMS, which investigate the fundamental particles and forces of the universe; ALICE, which studies the properties of quark-gluon plasma; and LHCb, which explores the differences between matter and antimatter. Through these experiments CERN generates petabytes of data that drive cutting-edge research in particle physics, pushing the boundaries of human knowledge. Supporting this ambitious mission requires a robust and scalable IT infrastructure to manage the vast and varied data demands of scientists and engineers across the globe.

Aiding this effort is the Database on Demand service, which provides reliable, scalable, and easy-to-use database solutions for CERN's diverse community of users. This service enables researchers and engineers to rapidly create and manage databases without the need for extensive technical expertise, ensuring they can focus on their primary scientific objectives. It integrates seamlessly with CERN's infrastructure, offering high performance, security, and flexibility to support a wide range of applications, from data analysis to operational tasks.

This brief keynote delves into the role of PostgreSQL in CERN's Database on Demand service. PostgreSQL's advanced features and open-source ethos align seamlessly with CERN's collaborative and innovative spirit. It shows the importance of collaboration between scientific innovation and open-source technologies in driving discoveries that benefit humanity.