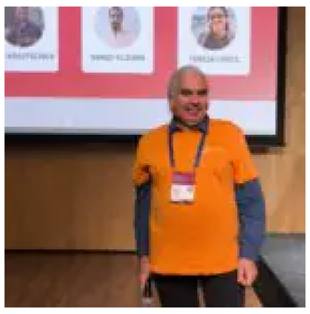
**10 NOVEMBER 2025**

# Representing credativ at PostgreSQL Conference Europe 2025

The European PostgreSQL Conference (PGConf.EU) is one of the largest PostgreSQL events worldwide. In this year it was held 21–24 October in Riga, Latvia. Our company, **credativ GmbH**, was a bronze sponsor of the conference, and I had the privilege to represent credativ with my talk "Database in Distress: Testing and Repairing Different Types of Database Corruption." In addition, I volunteered as a session host on Thursday and Friday. The conference itself covered a wide range of PostgreSQL topics – from cloud-native deployments to AI integration, from large-scale migrations to resiliency. Below are highlights from sessions I attended, organised by day.

## My talk about database corruption

I presenting my talk on Friday afternoon. In it I dove into real-world cases of PostgreSQL database corruption I encountered over the past two years. To investigate these issues, I built a Python tool that deliberately corrupts database pages and then examined the results using PostgreSQL's pageinspect extension. During the talk I demonstrated various corruption scenarios and the errors they produce, explaining how to diagnose each case. A key point was that PostgreSQL 18 now enables data checksums by default at initdb. Checksums allow damaged pages to be detected and safely "zeroed out" (skipping corrupted data) during recovery. Without checksums, only pages with clearly corrupted headers can be automatically removed using the zero_damaged_pages = on setting. Other types of corruption require careful manual salvage. I concluded by suggesting improvements (in code or settings) to make recovery easier on clusters without checksums.



## Tuesday: Kubernetes and AI Summits

Tuesday began with two half-day Summits. The PostgreSQL on Kubernetes Summit explored running Postgres in cloud-native environments. Speakers compared Kubernetes operators (CloudNativePG, Crunchy, Zalando, etc.), backup/recovery in Kubernetes, scaling strategies, monitoring, and zero-downtime upgrades. They discussed operator architectures and multi-tenant DBaaS use cases. Attendees gained practical insight into trade-offs of different operators and how to run Kubernetes-based Postgres for high availability.

In the PostgreSQL & AI Summit, experts examined Postgres's role in AI applications. Topics included vector search (e.g. pgvector), hybrid search, using Postgres as context storage for AI agents, conversational query interfaces, and even tuning Postgres with machine learning. Presenters shared best practices and integration strategies for building AIdriven solutions with Postgres. In short, the summit explored how PostgreSQL can serve AI workloads (and vice versa) and what new features or extensions are emerging for AI use cases.

## Wednesday: Migrations, Modelling, and Performance

Joaquim Oliveira (European Space Agency) discussed moving astronomy datasets (from ESA's Gaia and Euclid missions) off Greenplum. The team considered both scaling out with Citus and moving to EDB's new Greenplum-based cloud warehouse. He covered the practical pros and cons of each path and the operational changes required to re-architect such exascale workloads. The key lesson was planning architecture, tooling, and admin shifts needed before undertaking a petabyte-scale migration.

Boriss Mejias (EDB) emphasised that data modelling is fundamental to software projects. Using a chess-tournament application as an example, he showed how to let PostgreSQL enforce data integrity. By carefully choosing data types and constraints, developers can embed much of the business logic directly in the schema. The talk demonstrated "letting PostgreSQL guarantee data integrity" and building application logic at the database layer.

Roberto Mello (Snowflake) reviewed the many optimizer and execution improvements in Postgres 18. For example, the planner now automatically eliminates unnecessary self-joins, converts IN (VALUES…) clauses into more efficient forms, and transforms OR clauses into arrays for faster index scans. It also speeds up set operations (INTERSECT, EXCEPT), window aggregates, and optimises SELECT DISTINCT and GROUP BY by reordering keys and ignoring redundant columns. Roberto compared query benchmarks across Postgres 16, 17, and 18 to highlight these gains.

Nelson Calero (Pythian) shared a "practical guide" for migrating 100+ PostgreSQL databases (from gigabytes to multi-terabytes) to the cloud. His team moved hundreds of on-prem VM databases to Google Cloud SQL. He discussed planning, downtime minimisation, instance sizing, tools, and post-migration tuning. In particular, he noted challenges like handling old version upgrades, inheritance schemas, PostGIS data, and service-account changes. Calero's advice included choosing the right cloud instance types, optimising bulk data loads, and validating performance after migration.

Jan Wieremjewicz (Percona) recounted implementing Transparent Data Encryption (TDE) for Postgres via the pg_tde extension. He took the audience through the entire journey – from the initial idea, through patch proposals, to community feedback and design trade-offs. He explained why existing PostgreSQL hooks weren't enough, what friction was encountered, and how customer feedback shaped the final design. This talk served as a "diary" of what it takes to deliver a core encryption feature through the PostgreSQL development process.

Stefan Fercot (Data Egret) demonstrated how to use Patroni (for high availability) together with pgBackRest (for backups). He walked through YAML configuration examples showing how to integrate pgBackRest into a Patroni-managed cluster. Stefan showed how to rebuild standby replicas from pgBackRest backups and perform point-in-time recovery (PITR) under Patroni's control. The talk highlighted real-world operational wisdom: combining these tools provides automated, repeatable disaster recovery for Postgres clusters.

## Thursday: Cloud, EXPLAIN, and Resiliency

Maximilian Stefanac and Philipp Thun (SAP SE) explained how SAP uses PostgreSQL within Cloud Foundry (SAP's open-source PaaS). They discussed optimisations and scale challenges of running Postgres for SAP's Business Technology Platform. Over the years, SAP's Cloud Foundry team has deployed Postgres on AWS, Azure, Google Cloud, and Alibaba Cloud. Each provider's offerings differ, so unifying automation and monitoring across clouds is a major challenge. The talk highlighted how SAP contributes Postgres performance improvements back to the community and what it takes to operate large-scale, cloud-neutral Postgres clusters.

In "EXPLAIN: Make It Make Sense," Aivars Kalvāns (Ebury) helped developers interpret query plans. He emphasized that after identifying a slow query, you must understand why the planner chose a given plan and whether it is optimal. Aivars walked through EXPLAIN output and shared rules of thumb for spotting inefficiencies – for example, detecting missing indexes or costly operators. He illustrated common query anti-patterns he has seen in practice and showed how to rewrite them in a more database-friendly way. The session gave practical tips for decoding EXPLAIN and tuning queries.

Chris Ellis (Nexteam) highlighted built-in Postgres capabilities that simplify application development. Drawing on real-world use cases – such as event scheduling, task queues, search, geolocation, and handling heterogeneous data – he showed how features like range types, full-text search, and JSONB can reduce application complexity. For each use case, Chris demonstrated which Postgres feature or data type could solve the problem. This "tips & tricks" tour reinforced that leveraging Postgres's rich feature set often means writing less custom code.

Andreas Geppert (Zürcher Kantonalbank) described a cross-cloud replication setup for disaster resilience. Faced with a requirement that at most 15 minutes of data could be lost if any one cloud provider failed, they could not use physical replication (since their cloud providers don't support it). Instead, they built a multi-cloud solution using logical replication. The talk covered how they keep logical replicas up-to-date even as schemas change (noting that logical replication doesn't automatically copy DDL). In short, logical replication enabled resilient, low-RPO operation across providers despite schema evolution.

Derk van Veen (Adyen) tackled the deeper rationale behind table partitioning. He emphasised the importance of finding the right partition key – the "leading figure" in your data – and then aligning partitions across all related tables. When partitions share a common key and aligned boundaries, you unlock multiple benefits: decent performance, simplified maintenance, built-in support for PII compliance, easy data cleanup, and even transparent data tiering. Derk warned that poorly planned partitions can hurt performance terribly. In his case, switching to properly aligned partitions (and enabling enable_partitionwise_join/_aggregate) yielded a 70× speedup on 100+ TB financial tables. All strategies he presented have been battle-tested in Adyen's multi-100 TB production database.

## Friday: Other advanced Topics

Nicholas Meyer (Academia.edu) introduced thin cloning, a technique for giving developers real production data snapshots for debugging. Using tools like DBLab Engine or Amazon Aurora's clone feature, thin cloning creates writable copies of live data inexpensively. This lets developers reproduce production issues exactly – including data-dependent bugs – by debugging against these clones of real data. Nicholas explained how Academia.edu uses thin clones to catch subtle bugs early by having dev and QA teams work with near-production data.

Dave Pitts (Adyen) explained why future Postgres applications may use both B-tree and LSM-tree (log-structured) indexes. He outlined the fundamental differences: B-trees excel at point lookups and balanced reads/writes, while LSM-trees optimise high write throughput and range scans. Dave discussed "gotchas" when switching workloads between index types. The talk clarified when each structure is advantageous, helping developers and DBAs choose the right index for their workload.

A panel led by Jimmy Angelakos addressed “How to Work with Other Postgres People”. The discussion focused on mental health, burnout, and neurodiversity in the PostgreSQL community. Panelists highlighted that unaddressed mental-health issues cause stress and turnover in open-source projects. They shared practical strategies for a more supportive culture: personal “README” guides to explain individual communication preferences, respectful and empathetic communication practices, and concrete conflict resolution techniques. The goal was to make the Postgres community more welcoming and resilient by understanding diverse needs and supporting contributors effectively.

Lukas Fittl (pganalyze) presented new tools for tracking query plan changes over time. He showed how to assign stable Plan IDs (analogous to query IDs) so that DBAs can monitor which queries use which plan shapes. Lukas introduced the new pg_stat_plans extension (leveraging Postgres 18’s features) for low-overhead collection of plan statistics. He explained how this extension works and compared it to older tools (the original pg_stat_plans, pg_store_plans, etc.) and cloud provider implementations. This makes it easier to detect when a query’s execution plan changes in production, aiding performance troubleshooting.

Ahsan Hadi (pgEdge) described pgEdge Enterprise PostgreSQL, a 100% open-source distributed Postgres platform. pgEdge Enterprise Postgres provides built-in high availability (using Patroni and read replicas) and the ability to scale across global regions. Starting from a single-node Postgres, users can grow to a multi-region cluster with geo-distributed replicas for extreme availability and low latency. Ahsan demonstrated how pgEdge is designed for organizations that need to scale from single instances to large distributed deployments, all under the standard Postgres license.

## Conclusion

PGConf.EU 2025 was an excellent event for sharing knowledge and learning from the global PostgreSQL community. I was proud to represent credativ and to help as a volunteer, and I’m grateful for the many insights gained. The sessions above represent just a selection of the rich content covered at the conference. Overall, PostgreSQL’s strong community and rapid innovation continue to make these conferences highly valuable. I look forward to applying what I learned in my work and to attending future PGConf.EU events.

JM

ABOUT THE AUTHOR

## Josef Machytka

View posts →