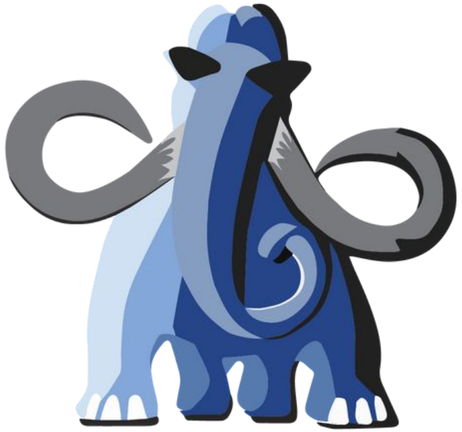
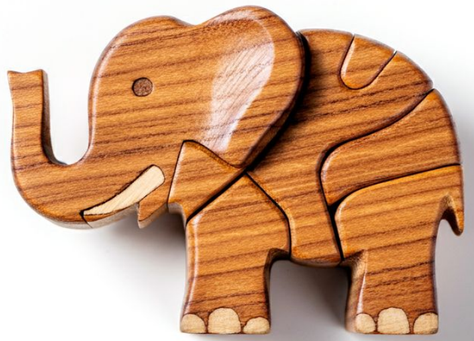
Journey Through Postgres:

Exploring the Elephant

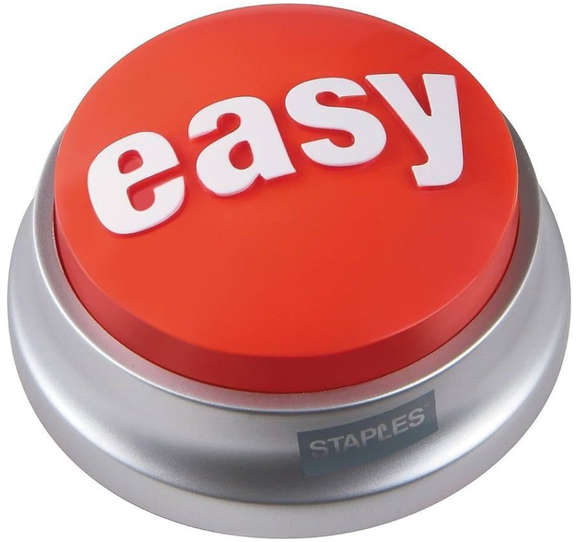


Presented by Zak Tedder

WHAT I THOUGHT THIS WOULD BE LIKE...



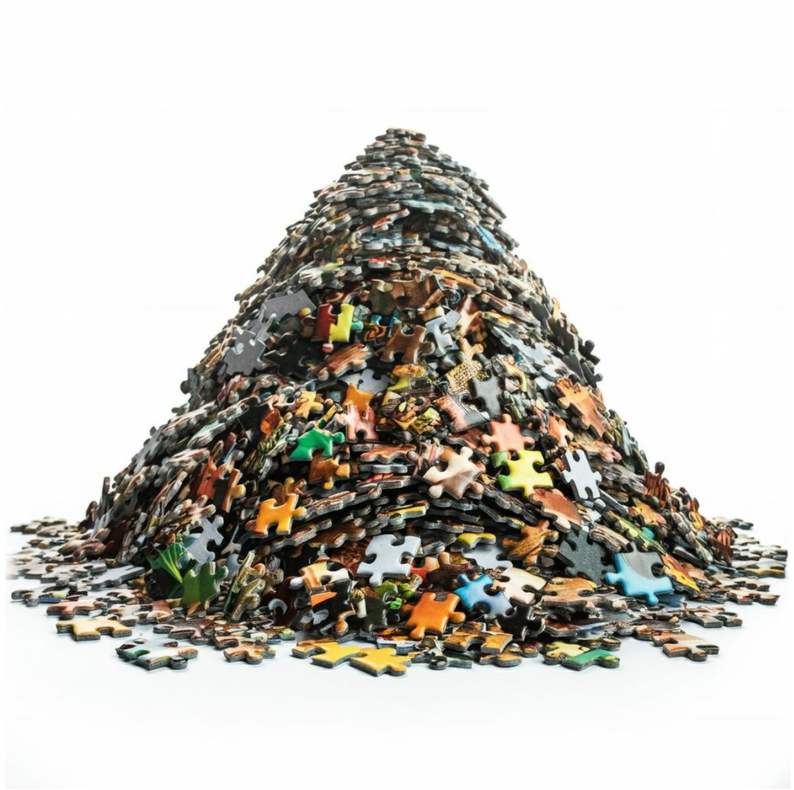
=



COMMAND PROMPT, INC.

Booth 630

WHAT THIS HAS TURNED OUT TO BE!



COMMAND PROMPT, INC.

Booth 630

WHAT I HOPE YOU'LL WALK AWAY WITH

COMMAND PROMPT, INC.

Booth 630

AGENDA

- Why PostgreSQL?
- Global Level
- Database Level
- Schema Level
- Table Level
- Internal Processes and Disk Structure
- Q&A
- More Resources

COMMAND PROMPT, INC.

Booth 630

WHY POSTGRESQL?

- Open source
- Strong community support
  - Active development
  - ACID compliant
  - Many advanced features
- Advanced indexing
  - Full-text search
- JSON support
- Cross-platform
- Customizability
  - Data types
  - Extensions
- Performance
- Scalability
- Robust Security

FOR MORE ON MIGRATIONS

Wednesday @ 11:40, Room 422 - Converting your schema from Oracle, MSSQL, MySQL to PostgreSQL in minutes with DataWharf - **Eric Lendvai**

Wednesday @ 2:00, Room 420 - PostgreSQL for Oracle DBAs - **Thuymy Tran & Chandra Pathivada**

Wednesday @ 3:00, Room 422 - DDL Schema Migrations: Navigating the High-Scale Seas: Learn about common failure modes when modifying Postgres tables and how to address them - **Sarah Munjal**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON MIGRATIONS

Wednesday @ 4:00, Room 420 - Bridging the Gap: Transition from SQL Server to Babelfish for Aurora PostgreSQL: Modernize your MSSQL Databases to Postgres with Babelfish - **Suraj Talreja**

Wednesday @ 4:00, Room 421 - Bridging the Gap: Smooth Data Migration from Oracle to PostgreSQL - **Clay Jackson**

Wednesday @ 5:00, Room 421 - Overcoming Migration Challenges: From Oracle to PostgreSQL - **Baji Shaik & Sameer Malik**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON MIGRATIONS

Thursday @ 11:40, Room 420 - Aurora PostgreSQL for SQL Server DBAs - **Thuymy Tran & Chandra Pathivada**

Thursday @ 12:10, Room 421 - Mastering RDS Maintenance: Strategies for Patching and Version Upgrades - **Arnab Saha**

Thursday @ 3:00, Room 422 - Seamless Transition: Migrating from Commercial to Open Source Databases: Using Aurora Postgres with Babelfish extension - **Alvaro Costa-Neto & Minesh Chande**

COMMAND PROMPT, INC.

Booth 630

GLOBAL LEVEL

COMMAND PROMPT, INC.

Booth 630

CLUSTERS

A PostgreSQL cluster is a collection of databases and their associated objects that exist on a server.

- A server can have multiple PostgreSQL clusters, including multiple versions
- Includes databases, tablespaces, roles, and configurations
- Global objects exist globally across the cluster

DATABASES

A database is a separate environment within a cluster that contains its own set of:

- Schemas
- Tables
- Indexes
- Sequences
- Extensions
- Data

TABLESPACES

Tablespaces are the physical storage locations for database objects.

- Can be associated with multiple databases
- Separate disk space for performance by creating dedicated storage space on specific drives

ROLES

Roles are database users and/or groups with specific permissions

- Control access to database objects
- Specify authentication configuration for specified roles
- Can configure specific settings for individual roles
  - Example: log_statement = 0

FOR MORE ON ROW-LEVEL SECURITY

Thursday @ 2:00, Room 422 - Efficient Row Level Security in Different Database Platforms: Design multi tenant solutions in data and generative AI using powerful knowledge bases - **Shailesh Doshi**

COMMAND PROMPT, INC.

Booth 630

LANGUAGES

Languages in PostgreSQL are the procedural languages used for writing functions and stored procedures. Examples include:

- Structured Query Language (SQL) - Core language for defining and manipulating data
- Procedural Language/PostgreSQL (PL/pgSQL) - Extends SQL with control structures, loops, and exception handling
- PL/Python - Enables function in Python
- PL/Perl - Enables function in Perl
- PL/Tcl - Enables functions in Tool Command Language (Tcl)

FOR MORE ON THE SQL LANGUAGE

Wednesday @ 2:00, Room 422 - Top 5 PostgreSQL Query Tuning Tips - **Janis Griffin**

Thursday @ 2:30, Room 420 - Query Optimizations and Where to Find Them: A Beginners Guide - **Ryan Booz**

COMMAND PROMPT, INC.

Booth 630

PHYSICAL REPLICATION

Physical replication replicates the actual data files from one location to another.

- Streams database changes via Write Ahead Log (WAL) files
- Hot Standby mode allows read queries on the standby while applying WALs
- Good for HA/DR setups
- Good for horizontal scaling to redirect read traffic

FOR MORE ON REPLICATION

Wednesday @ 10:40, Room 421 - Leveraging Patroni's synchronous replication feature to achieve high availability while running PostgreSQL on Kubernetes. - **Shree Vidhya Sampath**

Wednesday @ 10:40, Room 422 - A technical overview to unlock Seamless Logical Replication from Heterogeneous Databases (MySQL, Oracle, SQL Server, and More) to PostgreSQL - **Cary Huang**

Wednesday @ 11:10, Room 420 - FerretDB: Bringing MongoDB workloads to PostgreSQL - **Peter Farkas**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON REPLICATION

Wednesday @ 4:00, Room 422 - Architectural Considerations When Building Out Multi-Master Replication Clusters - **Robert Bernier**

Thursday @ 9:40, Room 422 - Time Travel Queries with Postgres - **Harry Pierson**

Thursday @ 10:40, Room 422 - Designing a check-in system for a space cruise line with PostgreSQL's new bidirectional replication - **Fernando Laudares Camargos**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON REPLICATION

Thursday @ 1:30, Room 421 - Designing Scalable Multi-Tenant Architectures with PostgreSQL for SaaS Solutions: Multi-tenancy with PostgreSQL databases - **Raj Jayakrishnan & Rajeev Thottathil**

Thursday @ 2:30, Room 422 - Overcoming Common Pitfalls of Postgres Logical Decoding - **Sai Srirampur**

Thursday @ 2:30, Room 421 - Kubernetes Killed the High Availability Star: How to stop worrying and embrace Postgres in the cloud - **Shaun Thomas**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON REPLICATION

Thursday @ 4:00, Room 421 - Managing A CitusDB Replication Cluster With Patroni - **Robert Bernier**

COMMAND PROMPT, INC.

Booth 630

CONFIGURATION

There are an extensive amount of settings in postgresql.conf that can be customized where needed.

- Can alter default settings via psql using ALTER SYSTEM SET, and setting will be added to the postgresql.auto.conf file
- Authentication settings are set in pg_hba.conf
- Settings can be customized at all levels; I will not be covering them

FOR MORE ON THE GLOBAL LEVEL

Wednesday @ 3:00, Room 420 - Professional PostgreSQL monitoring made easy - **Pavlo Golub**

Thursday @ 9:40, Room 421 - PostgreSQL 17: Performance and Efficiency - A Live Demo Showcase - **Kumar Ramamurthy & Adarsha Kuthuru**

COMMAND PROMPT, INC.

Booth 630

DATABASE LEVEL

COMMAND PROMPT, INC.

SCHEMAS

Schemas are logical containers within a database that organize objects.

- Provides a namespace for objects with the same name to exist in the same DB (public, archive, etc.)
- Queries can easily be created that work across schemas, but not databases or clusters

EXTENSIONS

Extensions are additional modules that enhance PostgreSQL functionality.

- Examples:
  - PostGIS
  - Timescale DB
  - pgcrypto
  - pg_stat_statements
  - pg_trgm
  - citext
  - Citus

**Note that you can create your own extensions!

FOR MORE ON EXTENSIONS

Wednesday @ 11:40, Room 420 - Leveling Up Your PostgreSQL Game using Extensions - **Anita Singh & Rajeev Thottathil**

Wednesday @ 11:40, Room 421 - Introducing SynchDB: Open-Source Synchronization Across Heterogeneous Databases with PostgreSQL: Seamless Integration for Modern Database Ecosystems and Bridging Data Silos - **Grant Zhou**

Wednesday @ 12:10, Room 422 - Beyond the Basics: Exploring PostgreSQL Catalogs and Extensions - **Swanand Kshirsagar & Venkataramanan Govindarajan**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON EXTENSIONS

Wednesday @ 5:30, Room 422 - Don't Stop Retrievin': Building lightning-fast search in Postgres - **Neil Hansen**

Thursday @ 11:40, Room 422 - Best practices for querying vector data for gen AI apps in PostgreSQL: PGvector extension - **Neelam Koshiya**

Thursday @ 3:00, Room 422 - Seamless Transition: Migrating from Commercial to Open Source Databases: Using Aurora Postgres with Babelfish extension - **Alvaro Costa-Neto & Minesh Chande**

COMMAND PROMPT, INC.

Booth 630

FUNCTIONS

Functions are a reusable block of SQL or procedural code that performs a specific task.

- Can intake parameter values, execute operations, and return a value
- Contains logic to reuse for regularly executed operations
- Can be used in SQL queries
- Aggregates perform calculations on a set of values (SUM, AVG, etc.)

PROCEDURES

Procedures are similar to functions, but do not return values.

- Typically used to perform operations, like modifying data
- Can have multiple transactions

DATA TYPES

PostgreSQL comes with standard data types.

- It also comes with a wide variety of other data types such as phone number, zip code, vector, and point
- You can create custom types in PostgreSQL, allowing for more complex structures and functionality

FOR MORE ON DATA TYPES

Wednesday @ 5:00, Room 422 - What's our Vector, Victor?: Taking the pain out of AI with pg_vectorize - **Shaun Thomas**

COMMAND PROMPT, INC.

Booth 630

LOGICAL REPLICATION

Logical replication replicates data at a higher level than physical replication.

- Can be targeted to replicate specific tables, data sets, or subsets of a table
- Uses publication/subscription setup
- Logically decodes WAL files to extract data changes and apply them to subscriber

FOR MORE ON REPLICATION

Wednesday @ 10:40, Room 421 - Leveraging Patroni's synchronous replication feature to achieve high availability while running PostgreSQL on Kubernetes. - **Shree Vidhya Sampath**

Wednesday @ 10:40, Room 422 - A technical overview to unlock Seamless Logical Replication from Heterogeneous Databases (MySQL, Oracle, SQL Server, and More) to PostgreSQL - **Cary Huang**

Wednesday @ 11:10, Room 420 - FerretDB: Bringing MongoDB workloads to PostgreSQL - **Peter Farkas**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON REPLICATION

Wednesday @ 4:00, Room 422 - Architectural Considerations When Building Out Multi-Master Replication Clusters - **Robert Bernier**

Thursday @ 9:40, Room 422 - Time Travel Queries with Postgres - **Harry Pierson**

Thursday @ 10:40, Room 422 - Designing a check-in system for a space cruise line with PostgreSQL's new bidirectional replication - **Fernando Laudares Camargos**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON REPLICATION

Thursday @ 1:30, Room 421 - Designing Scalable Multi-Tenant Architectures with PostgreSQL for SaaS Solutions: Multi-tenancy with PostgreSQL databases - **Raj Jayakrishnan & Rajeev Thottathil**

Thursday @ 2:30, Room 422 - Overcoming Common Pitfalls of Postgres Logical Decoding - **Sai Srirampur**

Thursday @ 2:30, Room 421 - Kubernetes Killed the High Availability Star: How to stop worrying and embrace Postgres in the cloud - **Shaun Thomas**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON REPLICATION

Thursday @ 4:00, Room 421 - Managing A CitusDB Replication Cluster With Patroni - **Robert Bernier**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON THE DATABASE LEVEL

Wednesday @ 3:00, Room 421 - Collation Surprises: Did Postgres Lose My Data? (2024) - **Jeremy Schneider**

Thursday @ 10:40, Room 420 - Collation Challenges: Sorting it Out - **Joe Conway**

COMMAND PROMPT, INC.

Booth 630

SCHEMA
LEVEL

COMMAND
PROMPT, INC.

TABLES

Tables are fundamental data storage structures where the actual data resides.

- Consist of rows and columns
- There are user tables, and system tables
- Types:
  - Regular
  - Partitioned
  - Inheritance
  - Foreign
  - TOAST
  - Temporary
  - Unlogged

PARTITIONED & INHERITANCE TABLES

Partitioned Tables are tables that are divided into smaller part(ition)s base on a column (i.e. date)

- Partitions make large amounts of data more manageable
- Can be used to simplify archiving data

Inheritance Tables are tables that inherit the structure and property of another table

- Allows for parent-child relationships
- Old method for partitioning (No longer recommended)

FOR MORE ON PARTITIONING

Wednesday @ 5:30, Room 420 - Mastering PostgreSQL Partitioning: Supercharge Performance and Simplify Maintenance - **Ryan Booz**

COMMAND PROMPT, INC.

Booth 630

FOREIGN TABLES

- Tables that reference data stored in another database or other external source
- Used with Foreign Data Wrapper (FDW) extension

TOAST TABLES

The Oversized-Attribute Storage Technique (TOAST) allows for the efficient storage of large data values that exceed a certain size (usually 2 KB)

- Helps manage oversized fields using text, bytea, and JSONB types
- Automatically compresses for storage and decompresses for data retrieval
- The main table keeps a pointer to the TOAST table
- Transparent to users of the main table, but not directly accessed/managed by users

SEQUENCES

Objects that generate unique numeric identifiers; not necessarily sequential.

- Primarily used for the iteration of Primary Keys
- Exist outside transactions, so safe to use for concurrent activity

VIEWS

Views are virtual tables created from querying one or more tables.

- Simplify complex queries
- There are user views, as well as system views
- Materialized Views are similar to views, but results in a physically stored data set
  - Must be updated manually

TRIGGERS

Triggers are functions that are called in response to a specific table event (like INSERT, UPDATE, DELETE)

- Event Trigger Functions respond to database-level events (DDL)
- These can exist at all different levels of the cluster

LOCKS

PostgreSQL requires Locks in some cases in order to perform certain operations while maintaining data integrity and consistency

Locks are managed automatically by PostgreSQL, but can also be explicitly acquired/released by users. There are a few types:

- Row-Level Locks
- Table-Level Locks
- Advisory Locks

ROW-LEVEL LOCKS

- Applied to individual rows and allow multiple transactions to operate on different rows of the same table simultaneously
- Deadlocks can occur if two transactions hold locks on different rows and each waits for the other to release its lock; PostgreSQL will abort one

	Current Lock Mode			
Requested Lock Mode	FOR KEY SHARE	FOR SHARE	FOR NO KEY UPDATE	FOR UPDATE
FOR KEY SHARE				X
FOR SHARE			X	X
FOR NO KEY UPDATE		X	X	X
FOR UPDATE	X	X	X	X

TABLE-LEVEL LOCKS

Table-level locks affect the entire table and prevent certain operations on the table while the lock is held

	Existing Lock Mode							
Requested Lock Mode	ACCESS SHARE	ROW SHARE	ROW EXCL.	SHARE UPDATE EXCL.	SHARE	SHARE ROW EXCL.	EXCL.	ACCESS EXCL.
ACCESS SHARE								X
ROW SHARE							X	X
ROW EXCL.					X	X	X	X
SHARE UPDATE EXCL.				X	X	X	X	X
SHARE			X	X		X	X	X
SHARE ROW EXCL.			X	X	X	X	X	X
EXCL.		X	X	X	X	X	X	X
ACCESS EXCL.	X	X	X	X	X	X	X	X

ADVISORY LOCKS

- User-defined and not tied to specific rows or tables
- Useful for application-level locking mechanisms
- Types:
  - Session-level: held until the session ends
    - Useful for long-running operations
  - Transaction-level: release at the end of a transaction
    - Useful for short-lived locks

LOCKING CONFLICTS

Deadlock detection:

- PostgreSQL automatically detects deadlocks and resolves them by aborting one transaction
- Typically occur with row-level locks when two transactions are waiting for each other to release

Lock waits:

- When a transaction requests a lock that is held by another transaction, it will wait. Can also configure an optional timeout for this
- Monitor with ‘pg_locks’ system view; wait events are also visible in pg_stat_activity

TABLE LEVEL

COMMAND PROMPT, INC.

CONSTRAINTS

Constraints are designed to maintain data integrity and ensure data adheres to specific rules. The tighter the rules, the higher level of integrity your data has. Balancing structure and flexibility is important.

- Types:
  - Primary Key
  - Foreign Key
  - Unique
  - Check
  - NOT NULL
  - Exclusion

PRIMARY KEY CONSTRAINTS

Primary Keys (PK) uniquely identify rows in a table

- They must contain unique values and may not be NULL
- One Primary Key per table, but it can be composed of multiple columns

FOREIGN KEY CONSTRAINTS

Foreign Key (FK) constraints create a link between two tables by referencing a column or multiple columns of another table (often the Primary Key)

- Backbone of a relational database!
- Ensure referential integrity
- Tables may have multiple FKs
- FKs can prevent deletes of referenced records; Cascade Delete deletes everything that depends on the record (CAUTION!!!: don’t use lightly)

UNIQUE CONSTRAINTS

A Unique constraint ensures that all non-null values in a column or set of columns are unique

- Unlike PKs, NULL values are allowed
- A table can have multiple Unique constraints

CHECK CONSTRAINTS

A Check constraint enforces specific conditions for a column's values or multiple columns' values

- The constraint checks whether a row meets a condition, and does not allow the data to be entered or updated if it does not
- A table can have multiple Check constraints
- Check constraints should not reference data from other rows

NOT NULL CONSTRAINTS

A NOT NULL constraint ensures that a column is not absent of value.

- Essentially a Check constraint, but specifically for a column to not have NULL
- This can be applied to any column in a table

EXCLUSION CONSTRAINTS

An Exclusion constraint ensures that a specified column does not have duplicate values when using a specific operator.

- Often used with geometric data types or ranges
- This can be applied to any column in a table
- Automatically creates an index to use

INDEXES

An object that organizes column values so that data is more quickly retrieved from the table.

- Types:
  - B-tree - Default; suitable for most queries
  - Hash - Used for equality comparisons
  - Generalized Inverted Index (GIN) - Full-text search and array data types
  - Generalized Search Tree (GiST) - Geometric data and range types
  - Block Range Index (BRIN) - For large tables with naturally ordered data

FOR MORE ON INDEXES

Wednesday @ 4:30, Room 420 - Why PostgreSQL Isn't Utilizing Indexes: Diagnosing and Solving Performance Issues - **Chandra Pathivada**

COMMAND PROMPT, INC.

Booth 630

VACUUMS

Vacuum is an operation that removes dead tuples in order to reclaim space for the table to reuse

- Autovacuum runs automatically based on thresholds
  - Settings are configurable
- Vacuum Full rewrites the table to a new disk location
  - Compacts table, reclaiming disk space
- Vacuum Analyze also

RULES AND TABLE SETTINGS

Rules are a feature that allows the definition of how certain SQL commands behave when applied to a table

- Can rewrite queries automatically, enabling complex behaviors without application changes

Some **Table Settings** can be modified on a per-table basis. For example:

- Compression
- Vacuum
- Fillfactor

INTERNAL PROCESSES & DISK STRUCTURE

COMMAND PROMPT, INC.

INTERNAL PROCESSES

- Postmaster - The main PostgreSQL process
- **Background Writer** - Writes dirty pages to disk
- Logical Replication Launcher - Handles starting subscription connections
- **Checkpointer** - Performs checkpoints
- Logger - Writes PostgreSQL logs when logging_collector is enabled
- **Statistics Collector** - Gathers stats about database activity (PostgreSQL < 15)
- **Autovacuum Launcher** - Starts autovacuum tasks

INTERNAL STRUCTURE ON DISK - SYSTEM LOGS

Logs are the various outputs generated by the database, which records information about operations

- Examples:
  - Error - Recorded Errors, Warnings, and other critical events
  - Query - Tracks executed SQL statements, including duration and execution stats
  - Connection - Records details about clients connections, disconnections, and failed attempts
  - Checkpoint - Documents occurrence of checkpoints, which impacts data durability
- Settings can be configured in postgresql.conf

INTERNAL STRUCTURE ON DISK - WAL

Write-Ahead Logging (WAL) files record changes made to the database before those changes are actually applied to the data files

- Ensures data integrity and durability, allowing for recovery in case of a crash
- WAL files are used for streaming replication, enabling redundancy
- When WAL files are written to disk, a checkpoint is created, helping to manage their size
- WAL files are crucial for point-in-time recovery (PITR)
- PostgreSQL can facilitate archiving WAL files that are no longer needed locally

FOR MORE ON POSTGRESQL INTERNALS

Wednesday @ 12:10, Room 421 - What makes PostgreSQL on Kubernetes interesting, even if it is maybe not for you? - **Fernando Laudares Camargos**

Wednesday @ 12:10, Room 420 - Practical Memory Tuning for PostgreSQL - **Grant McAlister**

Wednesday @ 2:00, Room 421 - Database DevOps: CD for Stateful Applications - **Stephen Atwell & Christopher Crow**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON POSTGRESQL INTERNALS

Thursday @ 10:40, Room 421 - Tackling Scheduling Challenges with pg_cron and pg_dbms_job - **Baji Shaik & Rajeshkumar Sabankar**

Thursday @ 11:40, Room 421 - Klutch: Revolutionizing PostgreSQL Automation with Kubernetes - **Julian Fischer**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON OTHER COOL ASPECTS

Thursday @ 9:40, Room 420 - SQL Team Six: How to Build Effective Teams: The essentials of leadership and effective teams - **Aaron Curshall**

Thursday @ 1:30, Room 422 - Innovating Customer Experience with Postgres: Transforming Health and Wellness at HERE Spa - **Bill Tang**

Thursday @ 2:00, Room 420 - Customizing the Wordle Game Experience with PostgreSQL - **Pavlo Golub**

COMMAND PROMPT, INC.

Booth 630

FOR MORE ON OTHER COOL ASPECTS

Thursday @ 2:30, Room 421 - Kubernetes Killed the High Availability Star: How to stop worrying and embrace Postgres in the cloud - **Shaun Thomas**

Thursday @ 3:30, Room 420 - Responsible AI in generative AI era: science and practice - **Neelam Koshiya**

COMMAND PROMPT, INC.

Booth 630

SUMMARY

- PostgreSQL is a huge, rapidly evolving ecosystem!
- We reviewed aspects at the Global Level, Database Level, Schema Level, Table Level, and Internal Processes
- Leveraging a framework that experience will help you navigate
- It can be complicated and messy, but there are resources out there:

Get in touch with Command Prompt to see how we can help optimize your database, and visit our learning resources:

www.commandprompt.com/education/

COMMAND PROMPT, INC.

Booth 630

WANT
MORE...?

COMMAND
PROMPT, INC.

FOR MORE...

Wednesday @ 10:40, Room 421 - Leveraging Patroni's synchronous replication feature to achieve high availability while running PostgreSQL on Kubernetes. - **Shree Vidhya Sampath**

Wednesday @ 10:40, Room 422 - A technical overview to unlock Seamless Logical Replication from Heterogeneous Databases (MySQL, Oracle, SQL Server, and More) to PostgreSQL - **Cary Huang**

Wednesday @ 11:10, Room 420 - FerretDB: Bringing MongoDB workloads to PostgreSQL - **Peter Farkas**

COMMAND PROMPT, INC.

Booth 630

FOR MORE...

Wednesday @ 11:40, Room 421 - Introducing SynchDB: Open-Source Synchronization Across Heterogeneous Databases with PostgreSQL: Seamless Integration for Modern Database Ecosystems and Bridging Data Silos - **Grant Zhou**

Wednesday @ 11:40, Room 422 - Converting your schema from Oracle, MSSQL, MySQL to PostgreSQL in minutes with DataWharf - **Eric Lendvai**

Wednesday @ 11:40, Room 420 - Leveling Up Your PostgreSQL Game using Extensions: PostgreSQL extensions - **Anita Singh & Rajeev Thottathil**

COMMAND PROMPT, INC.

Booth 630

FOR MORE...

Wednesday @ 12:10, Room 421 - What makes PostgreSQL on Kubernetes interesting, even if it is maybe not for you? - **Fernando Laudares Camargos**

Wednesday @ 12:10, Room 422 - Beyond the Basics: Exploring PostgreSQL Catalogs and Extensions - **Swanand Kshirsagar & Venkataramanan Govindarajan**

Wednesday @ 12:10, Room 420 - Practical Memory Tuning for PostgreSQL - **Grant McAlister**

COMMAND PROMPT, INC.

Booth 630

FOR MORE...

Wednesday @ 2:00, Room 422 - Top 5 PostgreSQL Query Tuning Tips - **Janis Griffin**

Wednesday @ 2:00, Room 420 - PostgreSQL for Oracle DBAs - **Thuymy Tran & Chandra Pathivada**

Wednesday @ 2:00, Room 421 - Database DevOps: CD for Stateful Applications - **Stephen Atwell & Christopher Crow**

COMMAND PROMPT, INC.

Booth 630

FOR MORE...

Wednesday @ 3:00, Room 420 - Professional PostgreSQL monitoring made easy - **Pavlo Golub**

Wednesday @ 3:00, Room 421 - Collation Surprises: Did Postgres Lose My Data? (2024): putting words in order without losing your mind or your data - **Jeremy Schneider**

Wednesday @ 3:00, Room 422 - DDL Schema Migrations: Navigating the High-Scale Seas: Learn about common failure modes when modifying Postgres tables and how to address them - **Sarah Munjal**

COMMAND PROMPT, INC.

Booth 630

FOR MORE...

Wednesday @ 4:00, Room 422 - Architectural Considerations When Building Out Multi-Master Replication Clusters - **Robert Bernier**

Wednesday @ 4:00, Room 420 - Bridging the Gap: Transition from SQL Server to Babelfish for Aurora PostgreSQL: Modernize your MSSQL Databases to Postgres with Babelfish - **Suraj Talreja**

Wednesday @ 4:00, Room 421 - Bridging the Gap: Smooth Data Migration from Oracle to PostgreSQL - **Clay Jackson**

COMMAND PROMPT, INC.

Booth 630

FOR MORE...

Wednesday @ 4:30, Room 420 - Why PostgreSQL Isn't Utilizing Indexes: Diagnosing and Solving Performance Issues - **Chandra Pathivada**

Wednesday @ 5:00, Room 421 - Overcoming Migration Challenges: From Oracle to PostgreSQL - **Baji Shaik & Sameer Malik**

Wednesday @ 5:00, Room 422 - What's our Vector, Victor?: Taking the pain out of AI with pg_vectorize - **Shaun Thomas**

COMMAND PROMPT, INC.

FOR MORE...

Wednesday @ 5:30, Room 422 - Don't Stop Retrievin': Building lightning-fast search in Postgres - **Neil Hansen**

Wednesday @ 5:30, Room 420 - Mastering PostgreSQL Partitioning: Supercharge Performance and Simplify Maintenance - **Ryan Booz**

COMMAND PROMPT, INC.

FOR MORE...

Thursday @ 9:40, Room 421 - PostgreSQL 17: Performance and Efficiency - A Live Demo Showcase - **Kumar Ramamurthy & Adarsha Kuthuru**

Thursday @ 9:40, Room 420 - SQL Team Six: How to Build Effective Teams: The essentials of leadership and effective teams - **Aaron Curshall**

Thursday @ 9:40, Room 422 - Time Travel Queries with Postgres - **Harry Pierson**

COMMAND PROMPT, INC.

FOR MORE...

Thursday @ 10:40, Room 420 - Collation Challenges: Sorting it Out - **Joe Conway**

Thursday @ 10:40, Room 421 - Tackling Scheduling Challenges with pg_cron and pg_dbms_job - **Baji Shaik & Rajeshkumar Sabankar**

Thursday @ 10:40, Room 422 - Designing a check-in system for a space cruise line with PostgreSQL's new bidirectional replication - **Fernando Laudares Camargos**

COMMAND PROMPT, INC.

FOR MORE...

Thursday @ 11:40, Room 422 - Best practices for querying vector data for gen AI apps in PostgreSQL: PGvector extension - **Neelam Koshiya**

Thursday @ 11:40, Room 420 - Aurora PostgreSQL for SQL Server DBAs - **Thuymy Tran & Chandra Pathivada**

Thursday @ 11:40, Room 421 - Klutch: Revolutionizing PostgreSQL Automation with Kubernetes - **Julian Fischer**

COMMAND PROMPT, INC.

FOR MORE...

Thursday @ 12:10, Room 421 - Mastering RDS Maintenance: Strategies for Patching and Version Upgrades - **Arnab Saha & Krishna Sarabu**

Thursday @ 1:30, Room 422 - Innovating Customer Experience with Postgres: Transforming Health and Wellness at HERE Spa - **Bill Tang**

Thursday @ 1:30, Room 421 - Designing Scalable Multi-Tenant Architectures with PostgreSQL for SaaS Solutions: Multi-tenancy with PostgreSQL databases - **Raj Jayakrishnan & Rajeev Thottathil**

COMMAND PROMPT, INC.

FOR MORE...

Thursday @ 2:00, Room 422 - Efficient Row Level Security in Different Database Platforms: Design multi tenant solutions in data and generative AI using powerful knowledge bases - **Shailesh Doshi**

Thursday @ 2:00, Room 420 - Customizing the Wordle Game Experience with PostgreSQL - **Pavlo Golub**

COMMAND PROMPT, INC.

FOR MORE...

Thursday @ 2:30, Room 422 - Overcoming Common Pitfalls of Postgres Logical Decoding - **Sai Srirampur**

Thursday @ 2:30, Room 420 - Query Optimizations and Where to Find Them: A Beginners Guide - **Ryan Booz**

Thursday @ 2:30, Room 421 - Kubernetes Killed the High Availability Star: How to stop worrying and embrace Postgres in the cloud - **Shaun Thomas**

COMMAND PROMPT, INC.

FOR MORE...

Thursday @ 3:00, Room 422 - Seamless Transition: Migrating from Commercial to Open Source Databases: Using Aurora Postgres with Babelfish extension - **Alvaro Costa-Neto & Minesh Chande**

Thursday @ 3:30, Room 420 - Responsible AI in generative AI era: science and practice - **Neelam Koshiya**

Thursday @ 4:00, Room 421 - Managing A CitusDB Replication Cluster With Patroni - **Robert Bernier**

COMMAND PROMPT, INC.