Learning PostgreSQL as a SQL Server User

Grant Fritchey

redgate | About me

Grant Fritchey

DevOps Advocate

Microsoft PostgreSQL MVP
AWS Community Builder

@gfritchey
Grant.Fritchey@Red-Gate.com
/in/gfritchey
scarydba.com

Agenda

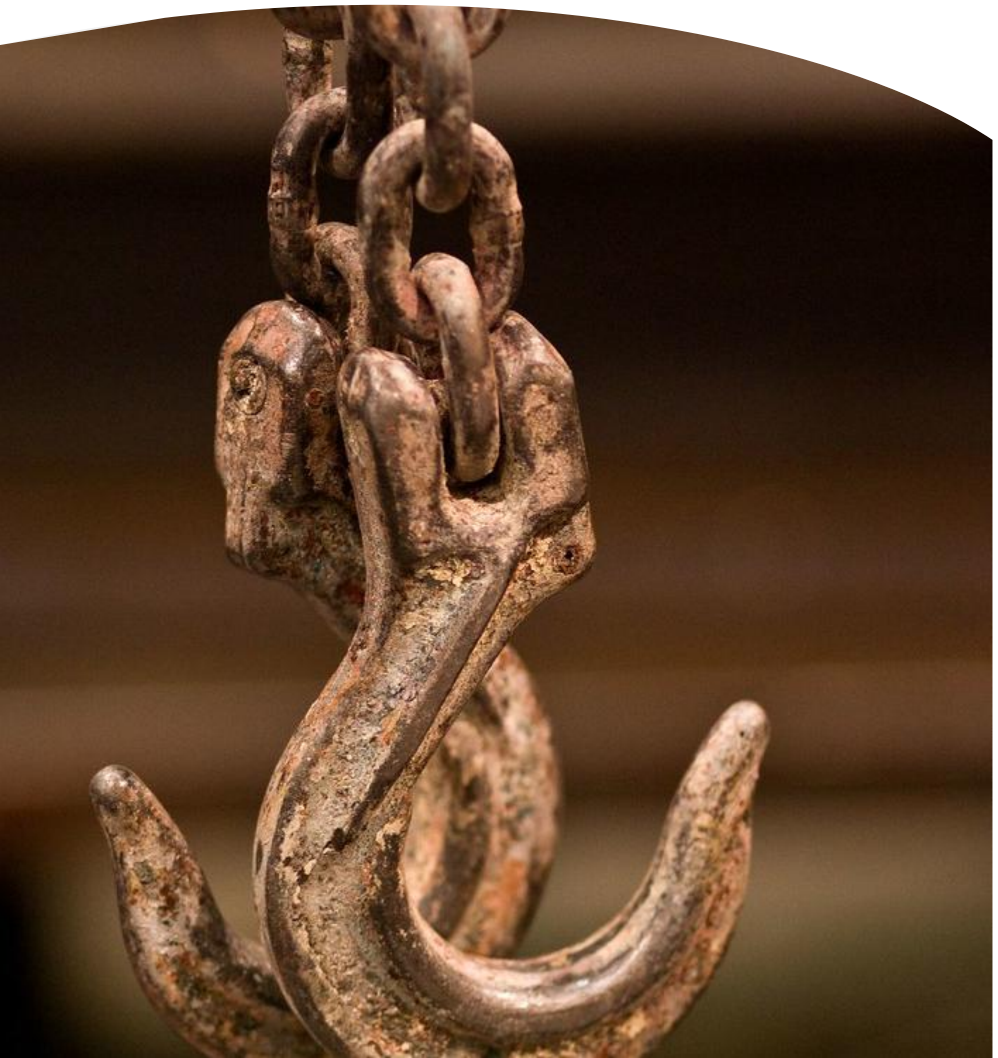
- Why PostgreSQL?
- A new language
- What's the same?
- What's different?
- What's confusing?
- How to go about learning PostgreSQL
- Where to got for more

Why PostgreSQL?

Licensing

- Extremely permissive, open-source license
- Core features can never be paywalled
- Most extensions piggyback on the PostgreSQL License

Extensibility



- >30 hooks available to extensions
- Any supported language, although SQL, C, and Rust are most popular

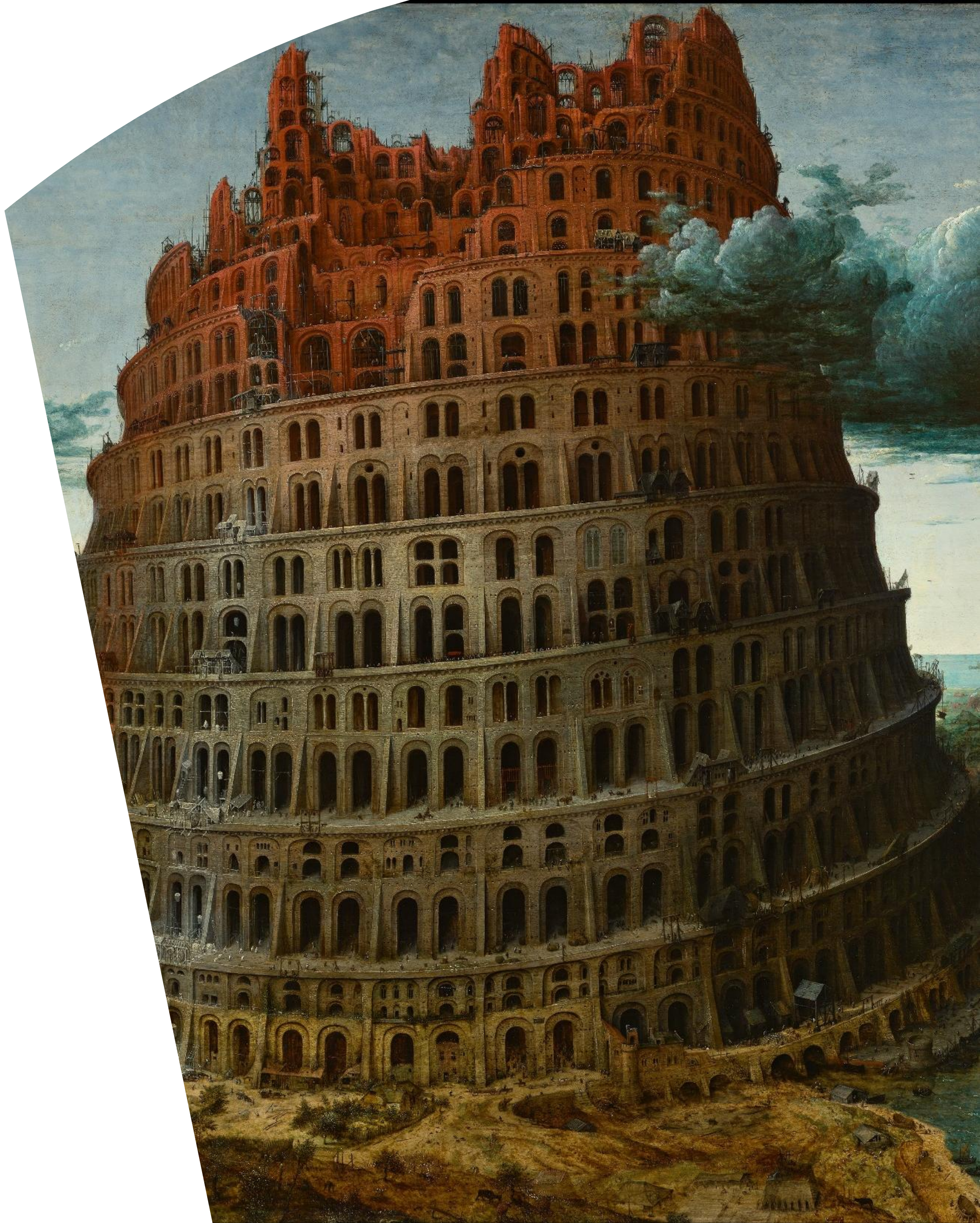
Extensibility

- Not just for extending functionality
- Package commonly used SQL functions/procedures
- Versioning required and helpful

A New Language

Confusion

- Words do not have the same meaning
- Take the time to learn the new terms
- It makes learning everything else easier



Mapping Terms

SQL Server	PostgreSQL
Instance	Cluster

Mapping Terms

SQL Server	PostgreSQL
User	Role

Mapping Terms

SQL Server	PostgreSQL
Transaction Log	Write Ahead Log (WAL)

Mapping Terms

SQL Server	PostgreSQL
Row	Tuple

Mapping Terms

SQL Server	PostgreSQL
Procedure	Function

Mapping Terms

SQL Server	PostgreSQL
Function	Procedure

What's The Same?

Core Concepts

- Relational database management system (well, Object Relational)
- SQL
- Normalization

Common Across Both

- Database
- Schema
- Table
- Primary key
- Foreign key

Common Across Both

- Column
- View
- Index
- Trigger
- Constraint

Common Across Both

- Query
- SELECT
- INSERT…
- In fact, strict adherence to the ANSI SQL standard (mostly)

What's Different?

Case Rules

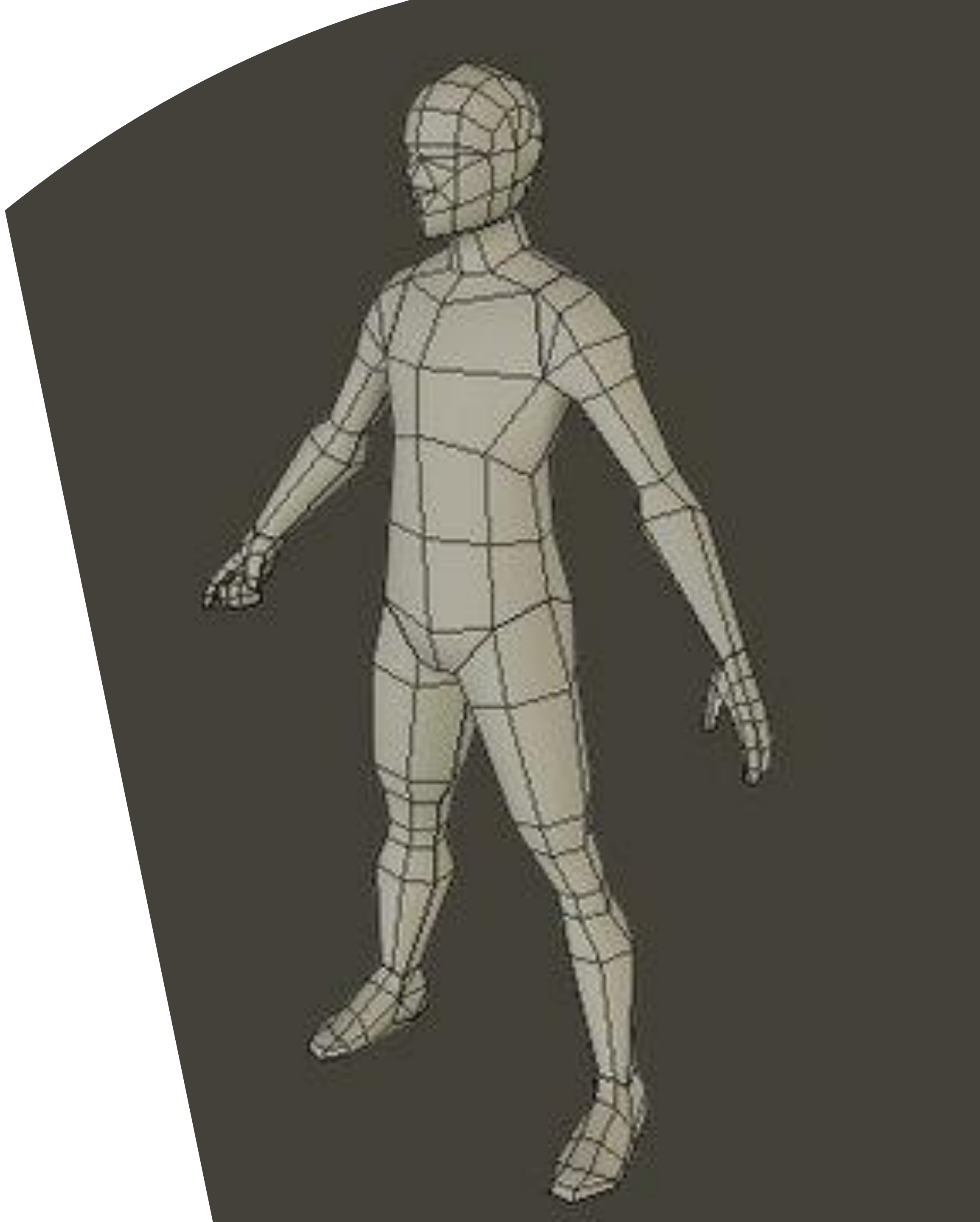
- All object names are lower case
- Unless you double “Quote” them
- commonly_use_snake_case

SQL

- LIMIT instead of TOP
- Ordinals instead of names
- DATE_PART instead of specific functions
- LATERAL instead of CROSS APPLY

Database Templates

- Think, model db
- template0 – leave this alone
- template1 – customizable
- Custom templates
- You can choose the template



CREATE DATABASE

- Defaults to template1
- You can pick the template
- You can define the owner

Backups & Restores

- pg_dump, pg_dumpall
- pg_basebackup
  - -- incremental (version 17+)
- WAL archiving for point-in-time recovery (more later)
- pg_restore
- Extensions

Data Types

- Twenty (20) Categories of data types
- Depending on how you count, ~125 data types

Data Type Behaviors

- Fixed or variable length, depending on type
- Most can be NULL
- DEFAULT values can be defined
- Can be treated as objects, even nested

Indexes

- PostgreSQL tables are heaps
- No concept of a clustered index
- CREATE and DROP CONCURRENTLY
- Must be table owner to add indexes

Index Types

- B-Tree
- Hash
- GiST
- SP-GiST
- GIN
- BRIN

Core Procedural PostgreSQL Languages

- SQL
- PL/pgSQL
- PL/Python
- PL/TcL
- PL/Perl

Server Metrics

- Error logs
- Cumulative Statistics System
    - Off by default
    - Reset on crash, failover, or point in time restore
    - Can be manually reset
    - Aggregations only

Cumulative Statistics System

- pg_stat_activity – current activity on server
- pg_stat_database – activity & size of database
- pg_stat_*_tables – activity & size of table
  - all
  - sys
  - user
- pg_stat_*_indexes – guess
- pg_statio_*_tables – specific info on blocks & storage

Query Metrics

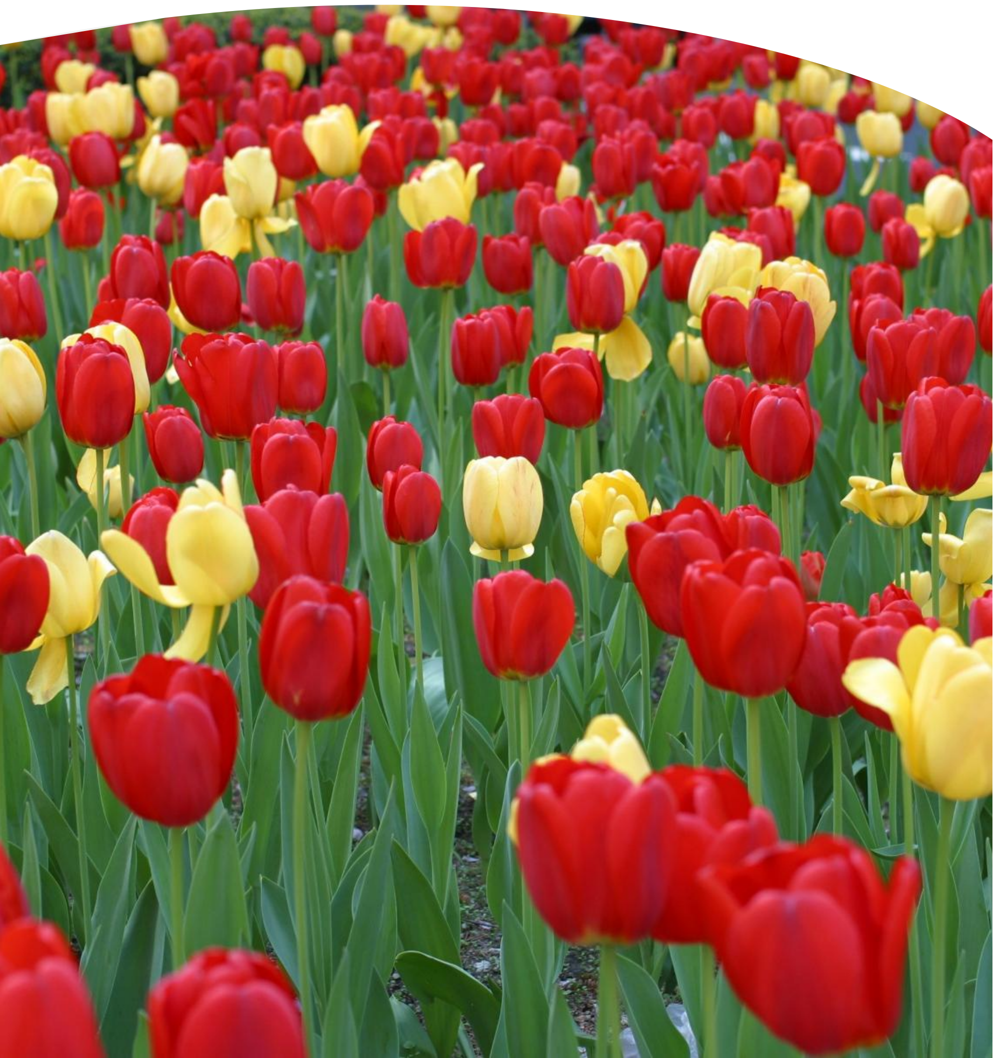
- EXPLAIN ANALYZE
- pg_stat_statements
  - Disabled by default

What's Confusing?

Restore to Point In Time

1. Run `pg_switch_wal`
2. Stop the server
3. Clean data directory or make a new one
4. Restore the Base Backup files
5. Remove all files from pg_wal/
6. Edit postgresql.conf for restore command and point in time
7. Create a file called recovery.signal in the cluster data directory
8. Start the service

Functions



- Implemented for returning data from PostgreSQL database
- Similar to Oracle functions or SQL Server procedures

Using Function Example

```
SELECT
   gl.genre_id,
   gl.genrename
FROM
genre_list() AS gl;
```

Procedures

- Procedures are meant to perform actions, as opposed to functions, which return data
- Similar to Oracle procedures, or SQL Server functions

Procedure Example

```
CREATE OR REPLACE PROCEDURE new_genre (genrename TEXT)
AS $$
BEGIN
   INSERT INTO bluebox.film_genre
     (name) VALUES (genrename);
END
$$
LANGUAGE plpgsql;
```

Vacuum

- Cleans dead tuples
- Reduces bloat
- Updates statistics

- Guaranteed, you need to run it more frequently

How To Learn PostgreSQL

Supported Operating Systems

- Linux
- Windows
- Containers
- Cloud

Packaged Bits

- Core
  - For the “hackers”
- Packages
  - OS
  - Special functions

Containers

- Easiest way to get started
- Get in the habit of setting up volumes
- Multiple forks available

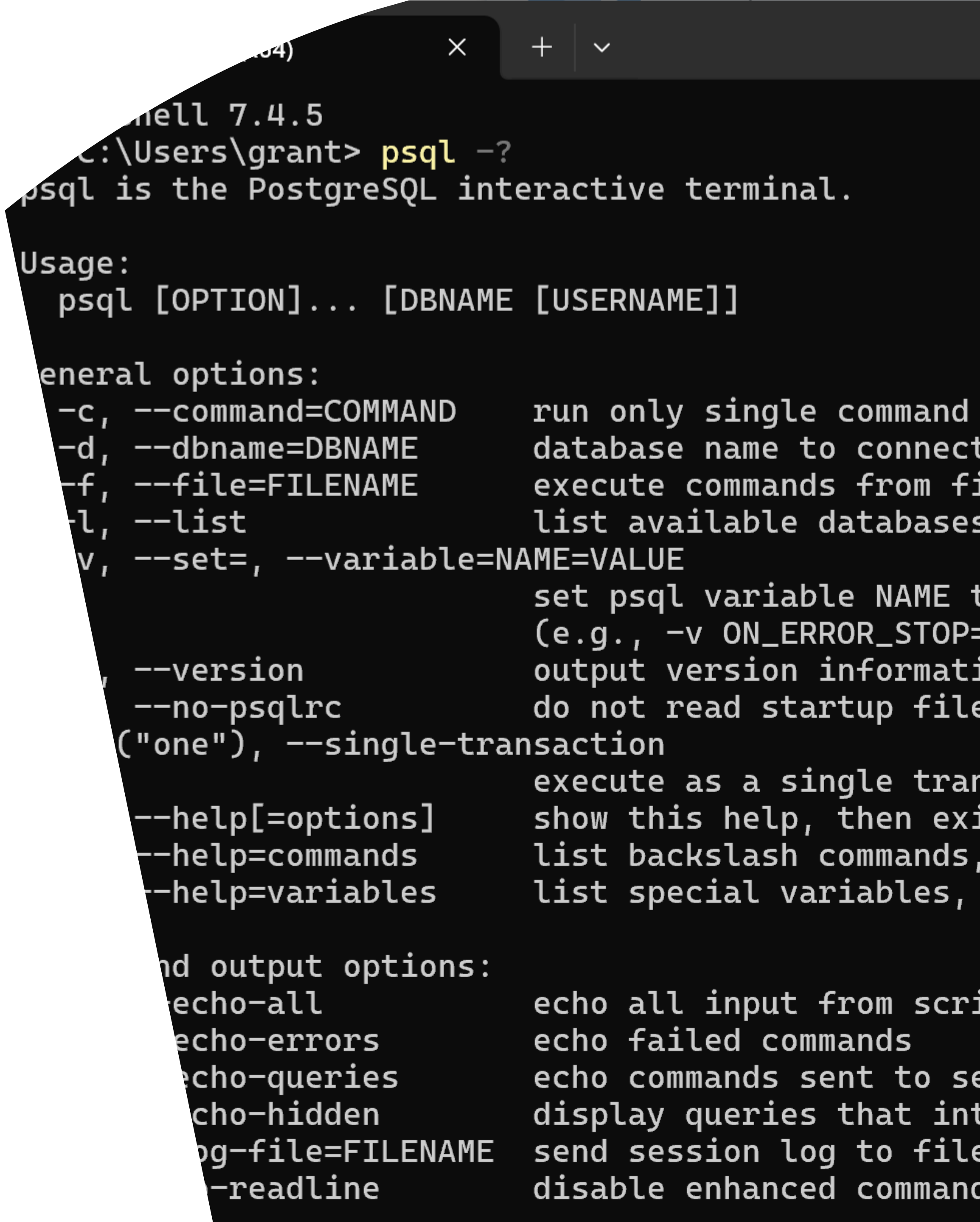
Cloud

- Azure
- Google Cloud
- AWS
- EnterpriseDB
- Heroku

PostgreSQL Tools

psql

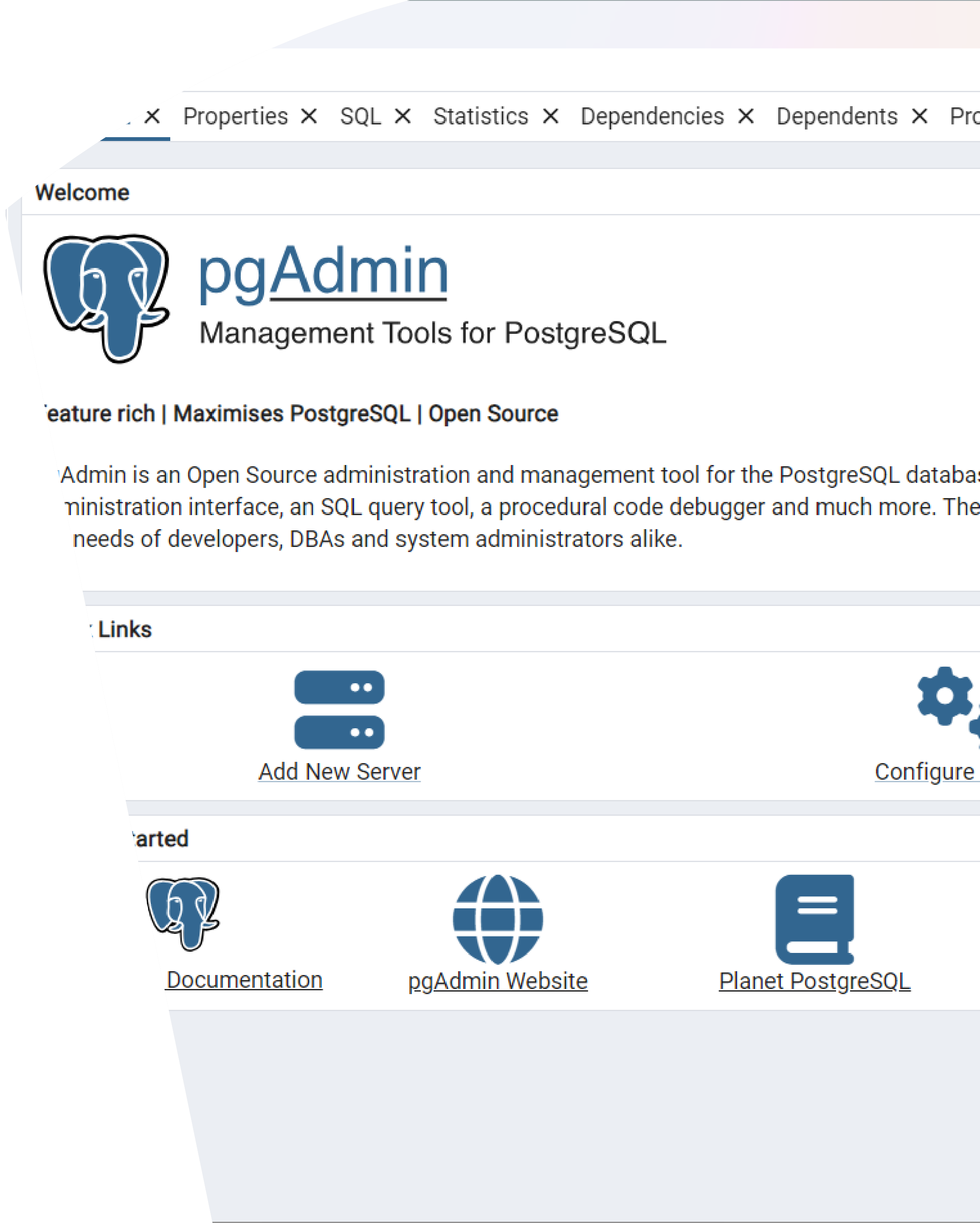
- Always available (almost)
- Command line
- Can work remote



```
                 ·64)
  ell 7.4.5
 C:\Users\grant> psql -?
psql is the PostgreSQL interactive terminal.

Usage:
  psql [OPTION]... [DBNAME [USERNAME]]

eneral options:
  -c, --command=COMMAND    run only single command
  -d, --dbname=DBNAME      database name to connect
  -f, --file=FILENAME      execute commands from fi
  -l, --list               list available databases
   v, --set=, --variable=NAME=VALUE
                           set psql variable NAME t
                           (e.g., -v ON_ERROR_STOP=
    , --version            output version informati
      --no-psqlrc          do not read startup file
     ("one"), --single-transaction
                           execute as a single tran
      --help[=options]     show this help, then exi
      --help=commands      list backslash commands,
      --help=variables     list special variables,

        nd output options:
        echo-all           echo all input from scri
        echo-errors        echo failed commands
        echo-queries       echo commands sent to se
        cho-hidden         display queries that int
        og-file=FILENAME   send session log to file
        -readline          disable enhanced command
```

pgAdmin

- Unofficial, "Official" PostgreSQL GUI
- Not installed by default
- Web version for remote work
- Extensions

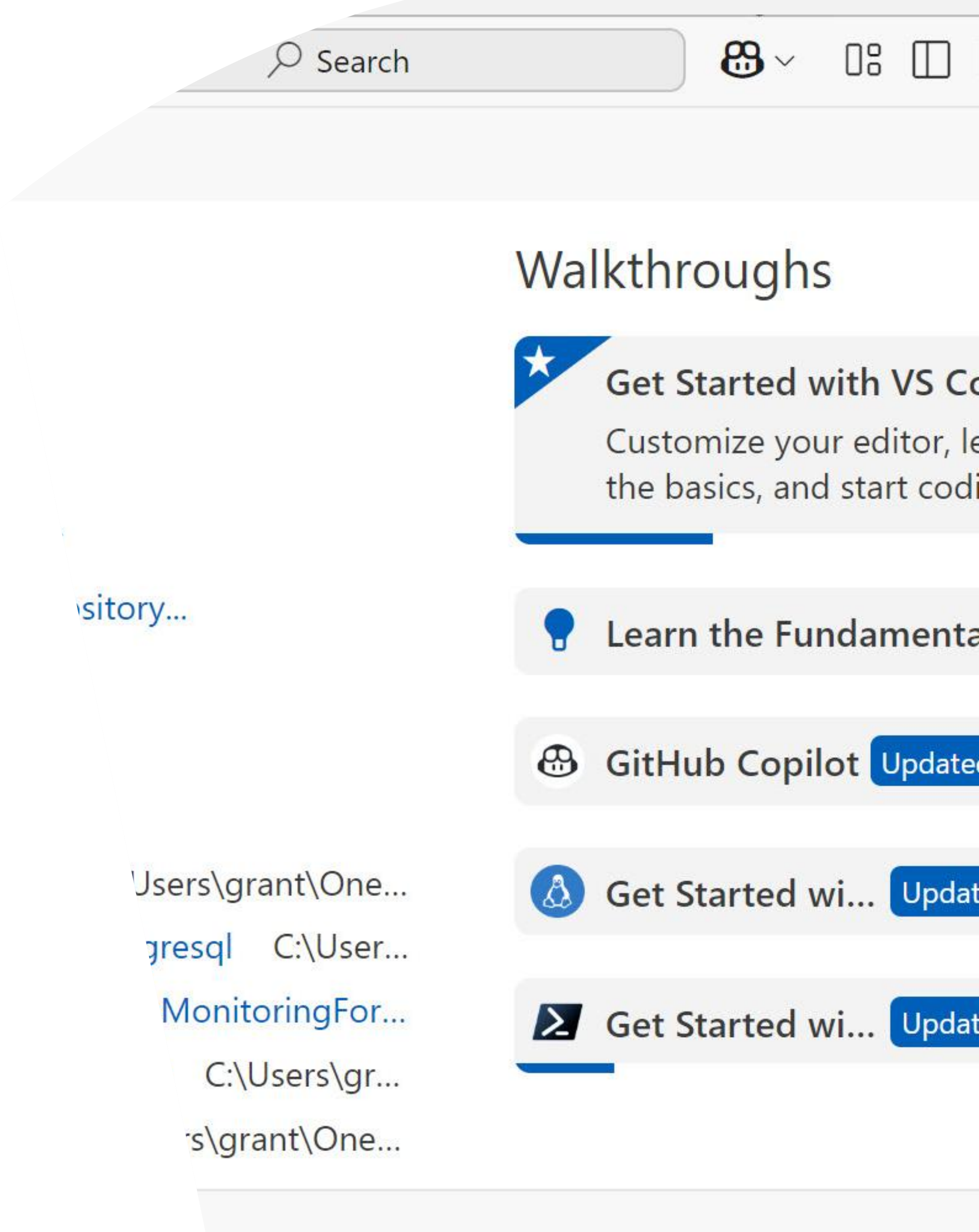


'eature rich | Maximises PostgreSQL | Open Source

'Admin is an Open Source administration and management tool for the PostgreSQL databa
ninistration interface, an SQL query tool, a procedural code debugger and much more. The
needs of developers, DBAs and system administrators alike.

VS Code

- PostgreSQL Plugins
- Good for basics
- Good integration with source control

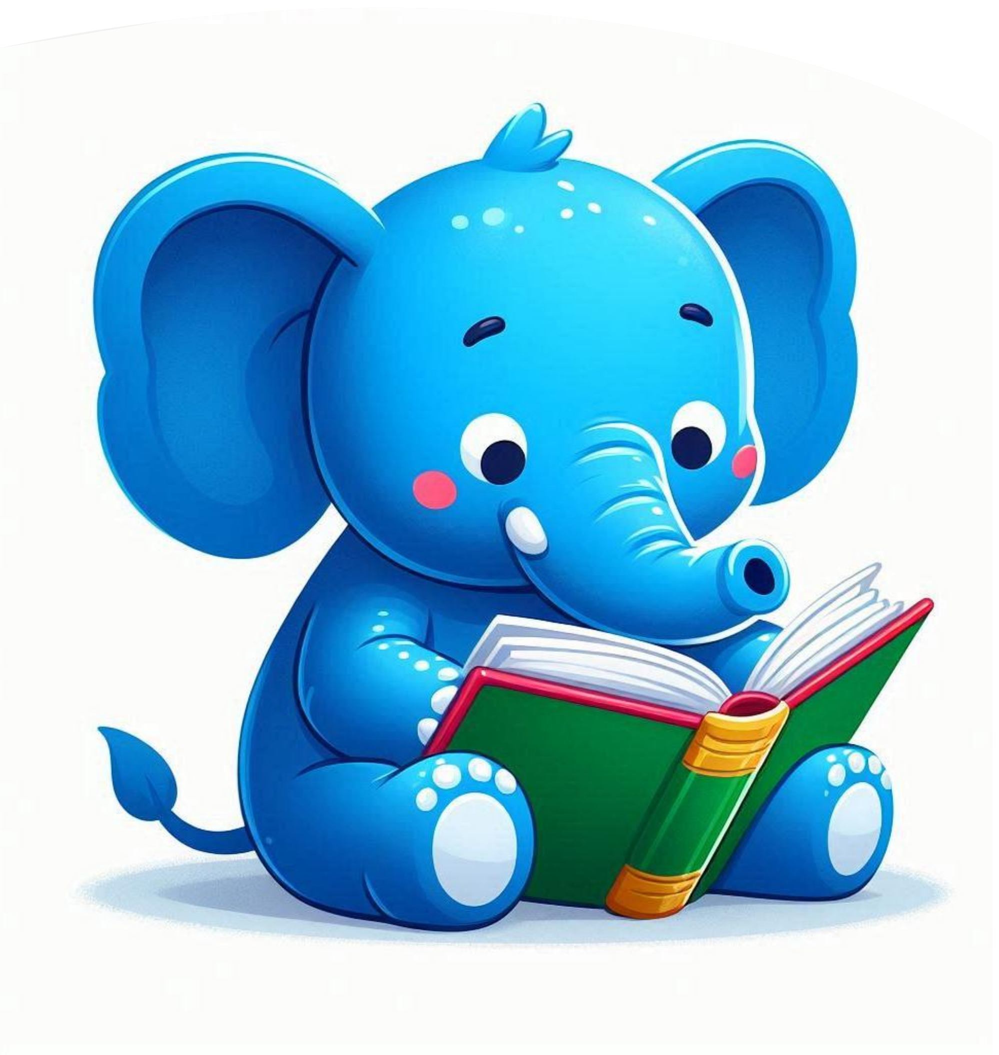


DBeaver

- Open source
  - Also paid
- Multi-platform
  - Good and bad
- Extensions
- Some code completion

More Learning

PostgreSQL Documentation



- https://www.postgresql.org/docs/
- Pay attention to this:
  - https://www.postgresql.org/docs/current/dml-delete.html

Books

- Introduction to PostgreSQL – Ryan Booz, Grant Fritchey
- Art of PostgreSQL – Dimitri Fontaine
- Database Administration – Craig Mullins
- PostgreSQL Query Optimization - Henrietta Dombrovskaya, Boris Novikov and Anna Bailliekova

Other Events

- PostgreSQL User Groups (PUG)
- MeetUp
- PGDay
- pgConf (same link as PGDay)

Online

- Planet PostgreSQL
- PostgreSQL Slack
- Cooper Press Email List
- #pghelp on X
- Usual suspects

Feedback!

Please!

redgate | About me

Grant Fritchey

DevOps Advocate

Microsoft PostgreSQL MVP
AWS Community Builder

@gfritchey
Grant.Fritchey@Red-Gate.com
/in/gfritchey
scarydba.com