

POSTGRESQL UPDATE

Neuigkeiten von PostgreSQL
und dessen Ökosystem

Michael Banck <michael.banck@netapp.com>
NetApp Deutschland GmbH
FrOSCon, 2024-08-17



POSTGRESQL - ÜBERBLICK

PostgreSQL - Überblick

- *“The World’s Most Advanced Open Source Relational Database”*
- Erweiterbares, Objekt-Relationales Datenbanksystem
- Entstanden als Forschungsprojekt in Berkeley, Community-basierte Entwicklung seit Mitte der 90er
- Hersteller-Unabhängig, kommerzieller Support von mehreren Firmen erhältlich
- “Postgres Global Development Group”, 7 köpfiges Core Team, ca. 30 Committer
- Keine Copyright-Assignments, Open-Core oder Dual-Lizenzierung
- BSD/MIT-artige Lizenz

Permission to use, copy, modify, and distribute this software and its documentation for any purpose, without fee, and without a written agreement is hereby granted, [...]

- Viele (auch proprietäre) Forks

PostgreSQL - Forks

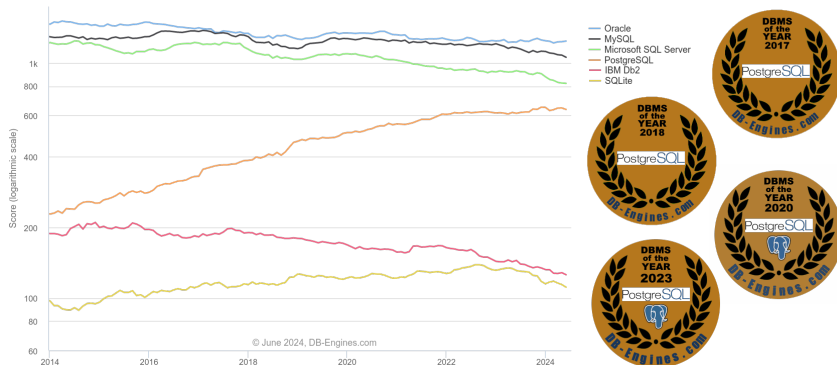


PostgreSQL - Forks



Popularity Contest

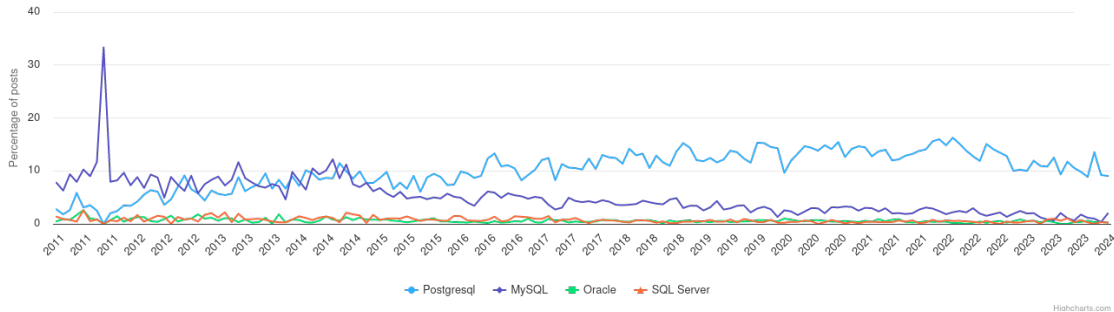
DB Engines Ranking



https://db-engines.com/en/ranking_trend/relational+dbms

Popularity Contest

Hacker News Hiring Trends



<https://www.hntrends.com/>

PostgreSQL Wire Protokoll

Die universelle Datenbank API

- Google Cloud Spanner Ankündigung bei der Google Cloud Next '21: * *“PostgreSQL has emerged as the “API” for operational databases”*



Cloud
Spanner



CockroachDB



Amazon Aurora



AlloyDB for PostgreSQL



CRATE.IO



yugabyteDB



ClickHouse



QuestDB



Materialize



Yellowbrick

Postlite

<https://dbdb.io/browse?compatible=postgresql>

PostgreSQL - Cloud Provider



PostgreSQL - Haupt-Features

- Datenintegrität, kompromisslos stabil und zuverlässig
- ACID-basierte Transaktionsverarbeitung
- Transaktionsisolation und -Nebenläufigkeit durch MVCC
- Transaktionale Änderungen an Datenbankstruktur
- Eingebaute Replikation, optional synchron für einzelne Transaktionen
- Gute und konsistente Abdeckung des SQL-Standards, sowie sinnvolle moderne Erweiterungen
- Kosten-basierter Query-Planer
- Hohe Anzahl an Treibern für Programmiersprachen sowie prozeduralen Sprachen
- Fremddaten-Wrapper (FDW) für den Zugriff auf viele andere Datenquellen/Datenbanken
- Große Anzahl an Erweiterungen

1



POSTGRESQL RELEASE CYCLE

PostgreSQL Major Release Cycle

- Eine Major-Version pro Jahr, Release üblicherweise in September/Oktober
- Zeit-basierter Code-Freeze (Q1), anschließende Beta/RC-Phase
- Release erfolgt wenn keine schweren Bugs mehr vorhanden sind
- 5 Jahre Support pro Release, finales Minor-Release danach

Version	Release Datum	Letztes Minor Release
16	14. September 2023	9. November 2028
15	13. Oktober 2022	11. November 2027
14	30. September 2021	12. November 2026
13	24. September 2020	13. November 2025
12	3. Oktober 2019	14. November 2024
11	18. Oktober 2018	9. November 2023
10	5. Oktober 2017	10. November 2022

PostgreSQL Minor Releases

- Major-Releases werden 5 Jahre lang supportet (sog. Backbranches)
- Vierteljährliche, planbare Point-Releases für kritische/sicherheitsrelevante Bugs
- Jeweils am zweiten Donnerstag des zweiten Monats eines Quartals
- Zeitplan: <https://www.postgresql.org/developer/roadmap/>
- Security Team behandelt Sicherheitsvorfälle
- Im Notfall außerordentliche Point-Releases
 - 14.4 Release am 16. Juni 2022
 - 10.3, 9.6.8, ... Release am 1. März 2018
- Distributions-Pakete für alle supporteten Versionen
 - Red Hat / CentOS / SLES: <http://yum.postgresql.org>, <http://zypp.postgresql.org>
 - Debian / Ubuntu: <http://apt.postgresql.org>

POSTGRESQL 15

PostgreSQL 15

- Release am 13.10.2022
- Aktuelle Version: 15.8
- Release-Announcement: <https://www.postgresql.org/about/news/postgresql-15-released-2526/>
- Release-Notes: <https://www.postgresql.org/docs/15/release-15.html>

```
grep '<listitem>' release-15.sgml | wc -l  
189
```


Breaking Changes

- Support für PostgreSQL 9.2 und früher von `pg_dump/psql` entfernt
- `public`-Schema hat keine `CREATE`-Rechte für alle Nutzer mehr
- Exklusiver Backup-Modus wurden entfernt, `pg_{start,stop}_backup()` → `pg_backup_{start,stop}()`
- Python2-Support entfernt

POSTGRESQL 16

PostgreSQL 16

- Release am 14.9.2023
- Aktuelle Version: 16.4
- Release-Announcement: <https://www.postgresql.org/about/news/postgresql-16-released-2715/>
- Release-Notes: <https://www.postgresql.org/docs/16/release-16.html>

```
$ grep '<listitem>' release-16.sgml | wc -l  
213
```

Breaking Changes

- Rollen mit CREATEROLE Rechten können nur noch selbst erstellte Rollen vergeben und verwalten
- `psql \du` Ausgabe zeigt keine Rollen-Mitgliedschaft mehr an (stattdessen nun `\drg`)
- `promote_trigger_file` Konfigurations-Parameter wurde entfernt
- `vacuum_defer_cleanup_age` Konfigurations-Parameter wurde entfernt
- Windows-Port supportet nur noch Windows 10 und neuer

Neue Features

Verbindungs-Kontrolle

- `pg_hba.conf` User und Datenbanken können Regexp's sein

```
host    /^dev/    all    10.1.0.0/24    scram-sha-256
```

- `pg_hba.conf` (und `pg_ident.conf`) erlaubt Einbinden von weiteren Dateien

```
local   all     all     peer
include_if_exists pg_hba.conf.d/test.conf
include pg_hba.conf.d/all.conf
```

- SCRAM Iterations-Zahl kann mit `scram_iterations` konfiguriert werden
- Kerberos Credential Delegation via `gss_accept_delegation` Parameter
 - Hilfreich für z.B. `postgres_fdw` und `dblink`
- Neue Server-Variable `SYSTEM_USER` für den zum Login verwendeten User, z.B. `peer:postgres`

Neue Features

Verbindungs-Optionen

- Load-Balancing bei mehreren Hosts: `load_balance_hosts=[random|disable]`
- `require_auth`, z.B. `require_auth=!password,!md5,scram-sha-256`
- `sslcertmode=[allow|disable|require]`
- kann System CA Store verwenden: `sslrootcert=system`
- Kerberos/GSS-Delegation `gssdelegation=[0|1]`

Neue Features

psql

- Kann nun das erweiterte Query-Protokoll via `\bind` verwenden

```
omdb=# SELECT p.name FROM people p WHERE p.id = $1 \bind 1 \gx
-[ RECORD 1 ]-----
name | George Lucas
```

- Neue Variablen `SHELL_ERROR` (t/f) und `SHELL_EXIT_CODE` für Shell-Befehle mit `\!`

Neue Features

Vacuum

- Autovacuum-Einstellungsänderungen werden praktisch sofort aktiv
 - Laufende Autovacuum-Skripts können beeinflusst werden
 - `ALTER TABLE foo SET (autovacuum_vacuum_cost_delay = 0ms);`
- Shared-Buffer Verbrauch von Vacuum kann nun via `vacuum_buffer_usage_limit` eingestellt werden
 - Bisher hartcodiert 256 KB, nun der Standardwert (2 MB ab V17)
 - Maximal 1/8 von Shared Buffers oder 16 GB
- Instrumentierung bei `VACUUM VERBOSE`/im Log verbessert, Prozent gefreezter Pages wird ausgegeben
- `VACUUM` kann optional nur die TOAST-Tabelle bearbeiten (`PROCESS_MAIN OFF`)
- `vacuumdb` Programm aktualisiert Statistiken in System-Tabellen nun nur noch am Ende

Neue Features

Berechtigungen

- Rollen mit CREATEROLE Rechten können nur noch selbst erstellte Rollen vergeben und verwalten

```
=# CREATE USER test CREATEROLE;  
CREATE ROLE  
=# SET ROLE test;  
SET  
=> GRANT pg_execute_server_program TO test;  
ERROR:  permission denied to grant role "pg_execute_server_program"  
DETAIL:  Only roles with the ADMIN option on role "pg_execute_server_program"  
         may grant this role.
```

- CREATEROLE-Nutzer benötigen Rollen-Mitgliedschaft WITH ADMIN TRUE um diese weiter geben zu können

Neue Features

Berechtigungen

- Neues Privileg SET für Rollen-Grants (standardmäßig TRUE)

```
=# CREATE USER test;  
CREATE ROLE  
=# CREATE GROUP test_group;  
CREATE ROLE  
=# GRANT test_group TO test WITH SET FALSE;  
GRANT ROLE  
=# SET SESSION AUTHORIZATION test;  
SET  
=> SET ROLE test_group;  
ERROR:  permission denied to set role "test_group"
```

Neue Features

Monitoring und Statistiken

- Neue Sicht `pg_stat_io`
- Aufsplittung von I/O in verschiedene Typen (Client Backend, Walsender, Background Writer) und Kontexte (normal, bulk{read,write}, vacuum)

```
omdb=# SELECT context, read_time, write_time, extend_time, hits, evictions, reuses
omdb=# FROM pg_stat_io WHERE backend_type = 'client backend' AND object = 'relation';
 context | read_time | write_time | extend_time | hits | evictions | reuses
-----+-----+-----+-----+-----+-----+-----
bulkread | 4310.902 |          0 |           |   853 |          0 | 541640
bulkwrite |          0 | 8028.566 | 271.347 | 1642454 |          0 | 1646012
normal   | 10413.354 | 395.596 | 39.992 | 96086025 | 132930 | 
vacuum   | 82.151 |          0 |          0 |    344 |          0 | 29938
(4 rows)
```

Neue Features

Monitoring und Statistiken

- pg_stat_*_{tables,indexes} Sichten zeigen jetzt den Zeitstempel der letzten Sequential/Index-Scans an

```
omdb=# SELECT relname, last_seq_scan, last_idx_scan FROM pg_stat_user_tables
omdb=# ORDER BY last_seq_scan ASC LIMIT 5;
```

relname	last_seq_scan	last_idx_scan
trailers	2024-08-16 15:23:40.561265+02	2024-08-16 15:27:15.952315+02
movie_countries	2024-08-16 15:23:40.561265+02	2024-08-16 15:27:15.952315+02
movie_abstracts_de	2024-08-16 15:23:40.561265+02	2024-08-16 15:27:15.952315+02
movie_languages	2024-08-16 15:23:40.561265+02	2024-08-16 15:27:15.952315+02
people_aliases	2024-08-16 15:23:40.561265+02	2024-08-16 15:27:15.952315+02

(5 rows)

Neue Features

Backup und Restore

- `pg_dump` kann Daten nun auch mit LZ4 oder Zstandard komprimieren
 - `pg_dump -Fc -Z lz4:5`

```
$ for format in gzip lz4 zstd zstd:8; do echo $format
>     time pg_dump -Fc -Z $format -f omdb.dump.$format omdb
>     ls -lh omdb.dump.$format | awk '{print $5}'; done
gzip
3.645 s
36M
lz4
0.672 s
57M
zstd
0.746 s
35M
zstd:8
2.042 s
31M
```

Neue Features

Logische Replikation

- Replication Origins
 - Erlaubt Replikation in eine Tabelle von mehreren Publishern
 - `CREATE SUBSCRIPTION [...] WITH (origin = none);`
 - "Bi-Directional Replication"
- Replikation von Standbys
 - Logical Decoding von Standbys möglich
 - Publication muss auf Primary erstellt werden
- Parallele Anwendung von logischen Changesets
 - `max_parallel_apply_workers_per_subscription` Parameter
- `pg_create_subscription`: Neue vordefinierte Rolle
 - Erlaubt die Erstellung von Subscriptions für Logische Replikation für nicht-Superuser
- `REPLICA IDENTITY FULL` Tabellen können nun Indexe verwenden um Zeilen zu identifizieren

Neue Features

Performance

- Parallelisierung von FULL und RIGHT Joins
- Parallele Aggregat-Funktionen (`string_all()`, `array_agg()`)
- Schnelleres Bulk-Loading mit COPY
- SIMD-Support (SSE2/NEON)
- Weitere Informationen zu verbesserter Query-Performance:

<https://www.citusdata.com/blog/2024/02/08/whats-new-in-postgres-16-query-planner-optimizer/>

POSTGRESQL 17

PostgreSQL 17

- Release TBD, vermutlich Q3/2024
- Aktuelle Version: 17 Beta 3
- Draft Release-Notes: <https://www.postgresql.org/docs/17/release-17.html>

```
$ grep '<listitem>' release-17.sgml | wc -l  
184
```

- Debian/Ubuntu-Pakete: deb <http://apt.postgresql.org/pub/repos/apt/> bookworm-pgdg main 17
- RHEL/Rocky/Alma-Pakete:
https://download.postgresql.org/pub/repos/yum/testing/17/redhat/rhel-9-x86_64/

Breaking Changes

- Adminpack Erweiterung wurde entfernt
- `snapshot_too_old` Konfigurations-Parameter/Feature wurde entfernt
- `db_user_namespace` Konfigurations-Parameter wurde entfernt
- Wartungs-Arbeiten verwenden nun einen sicheren `search_path`
- Viele Spalten in Statistik-Sicht `pg_stat_bgwriter` wurden nach `pg_stat_checkpoint` verschoben
- Die SLRU-Cache Namen in `pg_stat_slru` wurden umbenannt
- AIX-Support entfernt

Neue Features

Administration

- `transaction_timeout` Konfigurations-Parameter
 - Beendet Transaktionen (egal ob `idle` oder `active`) nach entsprechender Zeit
- Speicherverbrauch von Vacuum verbessert, 1 GB-Limit entfernt, mehrfache Index-Scans meist unnötig
- `pg_maintain` Default Rolle für `VACUUM`, `ANALYZE`, `REINDEX`, `REFRESH MATERIALIZED VIEW`, `CLUSTER`, und `LOCK TABLE`
- `allow_alter_system` Parameter verbietet `ALTER SYSTEM` Feature
- `event_triggers = [on|off]` Parameter schaltet Event Triggers an/aus
- Neue Event Trigger für `REINDEX` und Login
- `COPY` hat nun Fehlerbehandlung, z.B. `WITH (FORMAT csv, ON_ERROR 'ignore')`
- Eingebauter Locale-Provider `builtin` mit C/C.UTF-8 Locales, Future-Proof
- Neue LibPQ Funktion `PQChangePassword` um sicher Passwörter zu ändern

Neue Features

Administration

- Wait Event Beschreibung in neuer pg_wait_events Sicht

```
=# SELECT a.pid, a.state, a.wait_event_type, a.wait_event, w.description, a.query
-# FROM pg_stat_activity a, pg_wait_events w
-# WHERE state = 'active' AND a.wait_event = w.name \gx
-[ RECORD 1 ]----+-----
pid           | 126315
state         | active
wait_event_type | Timeout
wait_event    | PgSleep
description   | Waiting due to a call to pg_sleep or a sibling function
query         | SELECT pg_sleep(3600);
```

Neue Features

Partitionierung

- Partitionen können jetzt aufgesplittet und zusammengefasst werden
- Exklusiver Lock auf die beteiligten Tabellen während der Operation

```
=# ALTER TABLE pgbench_accounts MERGE PARTITIONS
-# (pgbench_accounts_1, pgbench_accounts_2) INTO pgbench_accounts_1_2;
ALTER TABLE
=# \d+ pgbench_accounts
[...]
Partitions: pgbench_accounts_1_2 FOR VALUES FROM (MINVALUE) TO (20000001),
            pgbench_accounts_3 FOR VALUES FROM (20000001) TO (30000001),
            pgbench_accounts_4 FOR VALUES FROM (30000001) TO (40000001),
            pgbench_accounts_5 FOR VALUES FROM (40000001) TO (MAXVALUE)
=# ALTER TABLE pgbench_accounts SPLIT PARTITION pgbench_accounts_1_2 INTO (
(# PARTITION pgbench_accounts_1 FOR VALUES FROM (MINVALUE) TO (10000001),
(# PARTITION pgbench_accounts_2 FOR VALUES FROM (10000001) TO (20000001));
ALTER TABLE
```

Neue Features

Monitoring und Statistiken

- Alle Spalten in `pg_stat_bgwriter` bis auf `buffers_clean`, `maxwritten_clean`, `buffers_alloc`, `stats_reset` wurden in neue Sicht `pg_stat_checkpoint` migriert
- `pg_stat_vacuum_progress` zeigt an wieviele Indexe bereits durchlaufen wurden
- `blk_{read,write}_time` in `pg_stat_statements` wurde zu `{local,shared}_blk_{read,write}_time` aufgesplittet
- Zusätzliche Spalte `{minmax_,}stats_since` in `pg_stat_statements`
- Normalisierung von Parametern in `CALL/SAVEPOINT`-Aufrufen in `pg_stat_statements`

Neue Features

Backup und Restore

- Inkrementelle Base Backups
- `summarize_wal = on / wal_summary_keep_time = '1d'`

```
$ pg_basebackup -D full/backup1; sleep 60 \  
> pg_basebackup --incremental=full/backup1/backup_manifest -D incr/backup2  
$ du -sh full/backup1 incr/backup2  
1.9G  full/backup1  
25M   incr/backup2
```

- `pg_combinebackup` erstellt valides Backup aus einem Voll- und Inkrementellen Backups

```
$ pg_combinebackup -o comb/backup3 full/backup1 incr/backup2  
$ du -sh comb/backup3  
1.9G  comb/backup3
```

Neue Features

Backup und Restore

- pg_dump/pg_restore Include/Exclude Listen können nun aus einer Datei gelesen werden via --filter
- Large Object Handling von pg_dump wurde verbessert

```
lotest=# SELECT lo_create(id) FROM generate_series(1,100000) AS id;
$ for version in 16 17; do echo -n "$version: "; /usr/bin/time -f '%E %Mk mem' \
> pg_dump --cluster $version/main -Fd -f lotest-$version.dir lotest
> ls -lh lotest-$version.dir/toc.dat | awk '{print $5 " " " $9}'; done
16: 3:49.95 381248k mem
94M lotest-16.dir/toc.dat
17: 3:49.20 114728k mem
3.4M lotest-17.dir/toc.dat
```

- pg_dump im Directory-Format mit --sync-method=syncfs unter Linux bei vielen Dateien deutlich schneller

```
$ /usr/bin/time -f '%E %Mk mem' pg_dump --cluster 17/main --sync-method=syncfs \
> -Fd -f lotest-17-syncfs.dir lotest
0:28.82 114900k mem
```


Neue Features

Logische Replikation

- Logische Replikation kann nun nach einem Failover fortgeführt werden
 - Synchronisation von Replikationsslots, `sync_replication_slots` Parameter
 - Von `CREATE/ALTER SUBSCRIPTION` aus konfigurierbar
- `pg_upgrade` migriert nun Subscriptions und logische Replikationsslots
- `pg_createsubscriber` Programm ermöglicht die Umwandlung eines physischen Standbys in einen logischen Replika
- Hash-Indexe können nun auch für die Identifikation von Datenzeilen bei Logischer Replikation verwendet werden

Neue Features

Performance

- Streaming/Vectored I/O API für Sequential Scans und ANALYZE
 - ca 20% Performance-Gewinn für ANALYZE
- SLRU Caches sind nun in einzelne “Banks” aufgeteilt, mit jeweils dediziertem Locking
- SLRU Cache Größen können einzeln konfiguriert werden
 - {commit_timestamp,multixact_{member,offset},notify,serializable,{sub,}transaction}_buffers
 - (change requires restart)
 - blks_hit/blks_read in pg_stat_slru-Sicht ermöglicht Abschätzung welche Caches geändert werden könnten
- Paralleles CREATE INDEX für BRIN-Indexe

POSTGRESQL 18

PostgreSQL 18

And Beyond

- Release TBD, vermutlich Q3/2025
- Aktuell in Entwicklung, 1/5 Commitfests absolviert
- Mögliche Features:
 - Logische Replikation - DDL/Sequenz-Replizierung und Conflict-Resolution
 - Import/Export von Statistiken (für Major Upgrades)
 - SQL Property Graph Queries (SQL/PGQ)
- Long-Term Roadmap:
 - 64-Bit Transaktions-IDs
 - Multithreading
 - Direct-IO
 - Transparent Data Encryption

NEUE ERWEITERUNGEN

pgvector

- Similarity/Nearest-Neighbor Search für Vektoren
- <https://github.com/pgvector/pgvector>
- Aktuelle Version 0.7.4
- Baustein für ML/AI
- HNSW und IVFFlat Indexe
- PostgreSQL Lizenz

pgvectorscale

- Auf pgvector aufbauend
- <https://github.com/timescale/pgvectorscale>
- Aktuelle Version 0.2.0
- StreamingDiskANN Index
- PostgreSQL Lizenz

Hydra Columnar

- Column-Oriented Storage Engine, Teil von HydraDB
- <https://github.com/hydradatabase/hydra/tree/main/columnar>
- Aktuelle Version 1.1.2
- Komprimierung, gut für Auslagerung von Daten zur Analyse, DML schwierig
- AGPL-3.0 Lizenz (Fork von Citus Columnar)

pg_tier

- Data Tiering nach S3
- https://github.com/tembo-io/pg_tier
- Aktuelle Version 0.0.4
- Einzelne Tabellen können ausgelagert werden, werden zu Fremdtabellen in S3
- Apache-2.0 Lizenz

pg_search

- Volltextsuche ähnlich wie Elasticsearch/OpenSearch, basierend auf Tantivy
- https://github.com/paradedb/paradedb/tree/main/pg_search
- Aktuelle Version 0.9.0 (Teil von ParadeDB)
- BM25 Index
- AGPL-3.0 Lizenz

pg_tde

- Experimentelle Erweiterung für Transparente Verschlüsselung
- https://github.com/Percona-Lab/pg_tde
- Aktuelle Version 1.0.0-Beta
- Storage Access Method für Verschlüsselung von Tabellen-Daten und WAL
- MIT Lizenz

pg_ivm

- Inkremental View Maintenance, fortlaufend aktualisierte Materialisierte Sichten
- https://github.com/sraoss/pg_ivm
- Aktuelle Version 1.9
- Erlaubt beim REFRESH von Matviews nur die Änderungen seit dem letzten Refresh einzupflegen
- Externe Erweiterung eines unter Review stehenden Postgres Patches
- PostgreSQL Lizenz

THANK YOU