POSTGRESQL ALS OPEN SOURCE

BERLIN 2025

HANS-JÜRGEN SCHÖNIG

CYBERTEC
POSTGRESQL SERVICES & SUPPORT

PROUD CONTRIBUTOR TO
PostgreSQL
the world's most advanced open source database


AUSTRIA (HQ)
CYBERTEC POSTGRESQL
INTERNATIONAL (HQ)
ESTONIA
CYBERTEC POSTGRESQL
NORDIC
SWITZERLAND
CYBERTEC POSTGRESQL SWITZERLAND
POLAND
CYBERTEC POSTGRESQL
POLAND
URUGUAY
CYBERTEC POSTGRESQL
SOUTH AMERICA
SOUTH AFRICA
CYBERTEC POSTGRESQL
SOUTH AFRICA

“FREEDOM HAS MANY DIFFICULTIES & DEMOCRACY IS NOT PERFECT, BUT WE HAVE NEVER HAD TO PUT A WALL UP TO KEEP OUR PEOPLE IN”

JOHN F. KENNEDY, 1963

“FREEDOM HAS MANY DIFFICULTIES & DEMOCRACY IS NOT PERFECT, BUT WE HAVE NEVER HAD TO PUT A WALL UP TO KEEP OUR PEOPLE IN”

JOHN F. KENNEDY, 1963

Passt gut zu Software

CLOSED SOURCE:
WAS KÜRZLICH PASSIERT IST

“AUDIT LOG DATEN KOPIEREN”

DATABASE COMPLIANCE UND AUDIT

1. Compliance ist ein großes Thema

- Gesetzliche Vorschriften
- Regulierung, Zertifizierung, Vorgaben

2. Aggregierung und Analyse von Audit Daten

- Klare Sache bei PostgreSQL
- Wie sieht es bei Closed Source aus?

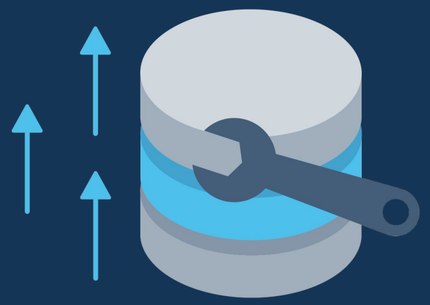
3. “Einige kleine Themen”

- Man beginnt die Unterschiede zu verstehen

WICHTIG

CLOSED SOURCE

WAS BEDEUTET DAS WIRKLICH?



Bugs und Fehler

zahlreich und störend

Knowledge Base

Was kann man eigentlich wissen?

Die Liebe zum Detail

Sinn vs. Gehaltsscheck

BUGS UND FEHLER

UND DER UMGANG DAMIT

EIN BEISPIEL

```
1| DBMS_AUDIT_MGMT.SET_LAST_ARCHIVE_TIMESTAMP(
2|     audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_UNIFIED,
3|     last_archive_time => TO_TIMESTAMP(:ts, 'YYYY-MM-DD HH24:MI:SS.FF')
4| );
5|
6| DBMS_AUDIT_MGMT.CLEAN_AUDIT_TRAIL(
7|     audit_trail_type => DBMS_AUDIT_MGMT.AUDIT_TRAIL_UNIFIED,
8|     use_last_arch_timestamp => true
9| );
```

- Bisher 4 verschiedene Fehlermeldungen …
- Idente Systeme mit identischer Last
- Ab und zu bleiben uralt Daten übrig

KNOWLEDGE BASE

WAS KANN MAN EIGENTLICH WISSEN?

“FRAGEN WIR MAL DEN GURU”

PostgreSQL Welt

- “Ich schau schnell nach”
- “Das macht …”
- “Ah, das kenne ich, wart kurz …”
- … ich hätte wohl gar nicht fragen müssen …

Kommerzielle Welt

- “Das weiß ich nicht”
- “Ich vermute mal, dass …”
- “… noch nie gesehen …”

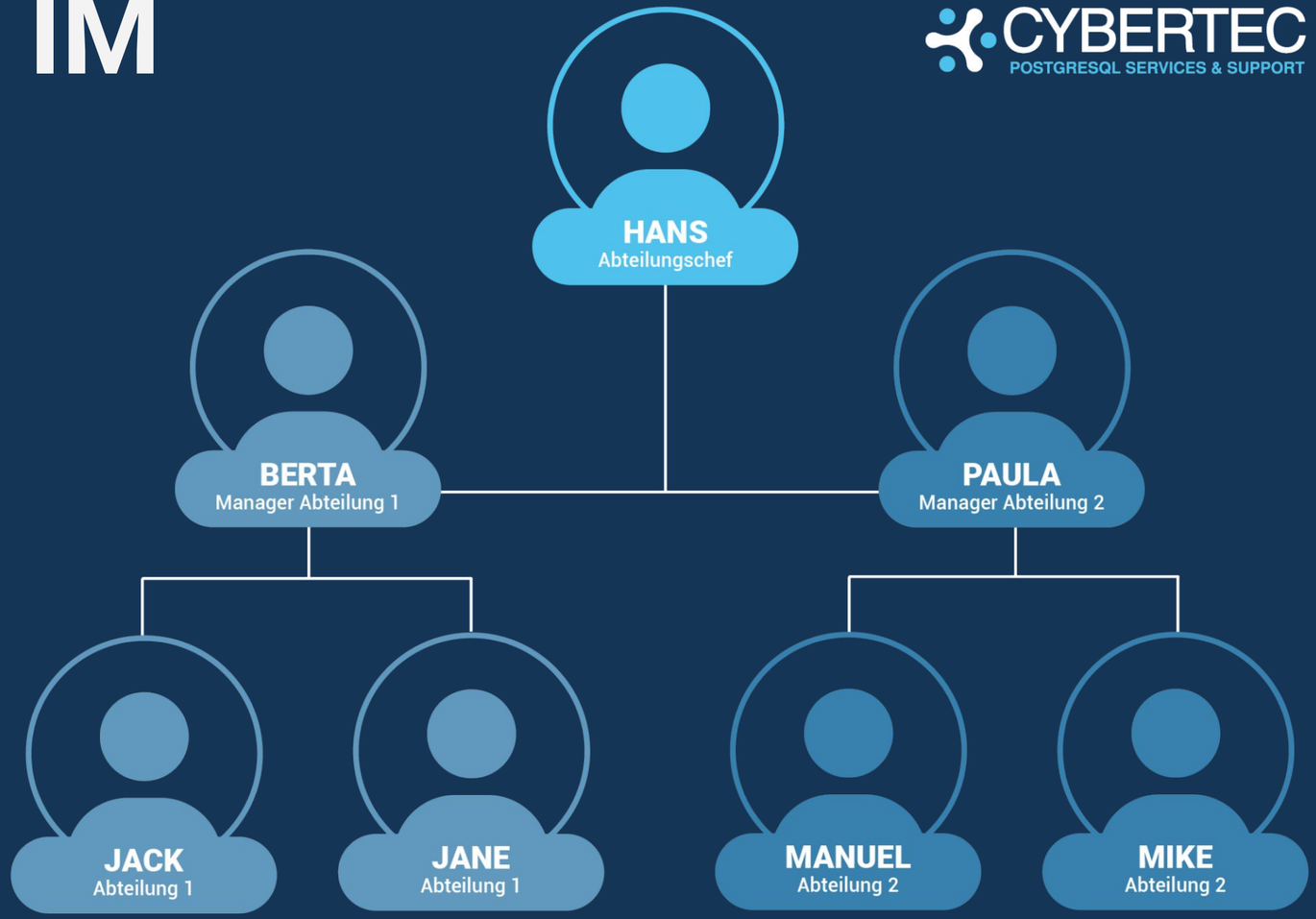
Problem

“ DIE ERKENNTNIS ”

WOHER SOLL DER KOLLEGE DAS EIGENTLICH WISSEN?

WAS HEISST DAS IM REALEN LEBEN?

- “Nichts wissen” = Research
- “Nichts wissen” = Verzögerung
- “Nichts wissen” = Risiko
- “Nichts wissen” = Potenziell falsch
- “Nichts wissen” = Frust
- “Nichts wissen” = Vertrauensverlust
- “Problem” = Ganz viele Leute in ganz vielen Meetings



ES GEHT BESSER

WAS HEISST OPEN SOURCE WIRKLICH

```
test=# SHOW effective_cache_size;
 effective_cache_size
----------------------
 4GB
(1 row)
```

Was tut das?

```
1| test=# SHOW effective_cache_size;
2|  effective_cache_size
3| ----------------------
4|  4GB
5| (1 row)
```

SEHEN WIR UNS DAS GENAUER AN...

```
1| hs@hans:~/src/postgresql-17.3/src/backend$ grep -r -n -I -l effective_cache_size *
2| access/gist/gistbuild.c
3| optimizer/path/costsize.c
4| utils/misc/postgresql.conf.sample
5| utils/misc/guc_tables.c
```

GIST? WAS PASSIERT DA?

```
731 |  /* subtree must fit in cache (with safety factor of 4) */
732 |     if (subtreesize > effective_cache_size / 4)
733 |         break;
```

OPTIMIZER? KLINGT SPANNEND…

```
 1| * costsize.c
 2|  *        Routines to compute (and set) relation sizes and path costs
 3| …
 4|  *      seq_page_cost           Cost of a sequential page fetch
 5|  *      random_page_cost        Cost of a non-sequential page fetch
 6|  *      cpu_tuple_cost          Cost of typical CPU time to process a tuple
 7|  *      cpu_index_tuple_cost  Cost of typical CPU time to process an index tuple
 8|  *      cpu_operator_cost       Cost of CPU time to execute an operator or function
 9|  *      parallel_tuple_cost Cost of CPU time to pass a tuple from worker to leader backend
10|  *      parallel_setup_cost Cost of setting up shared memory for parallelism
11| …
12|  * We also use a rough estimate "effective_cache_size" of the number of
13|  * disk pages in Postgres + OS-level disk cache.  (We can't simply use
14|  * NBuffers for this purpose because that would ignore the effects of
15|  * the kernel's disk cache.)
16|  *
17|  * Obviously, taking constants for these values is an oversimplification,
18|  * but it's tough enough to get any useful estimates even at this level of
19|  * detail.  Note that all of these parameters are user-settable, in case
20|  * the default values are drastically off for a particular platform.
```

OPTIMIZER? KLINGT SPANNEND...

```
 1|  * index_pages_fetched
 2|  *        Estimate the number of pages actually fetched after accounting for
 3|  *        cache effects.
 4|  *
 5|  * We use an approximation proposed by Mackert and Lohman, "Index Scans
 6|  * Using a Finite LRU Buffer: A Validated I/O Model", ACM Transactions
 7|  * on Database Systems, Vol. 14, No. 3, September 1989, Pages 401-424.
 8|  * The Mackert and Lohman approximation is that the number of pages fetched is
 9|  *      PF =
10|  *              min(2TNs/(2T+Ns), T)                        when T <= b
11|  *              2TNs/(2T+Ns)                                  when T > b and Ns <= 2Tb/(2T-b)
12|  *              b + (Ns - 2Tb/(2T-b))*(T-b)/T   when T > b and Ns > 2Tb/(2T-b)
13|  * where
14|  *    T = # pages in table,  N = # tuples in table
15|  *    s = selectivity = fraction of table to be scanned, b = # buffer pages available
16|  *
17|  * We assume that effective_cache_size is the total number of buffer pages
18|  * available for the whole query, and pro-rate that space across all the
19|  * tables in the query and the index currently under consideration.  (This
20|  * ignores space needed for other indexes used by the query, but since we
21|  * don't know which indexes will get used, we can't estimate that very well;
22|  * and in any case counting all the tables may well be an overestimate, since
23|  * depending on the join plan not all the tables may be scanned concurrently.)
```

OPTIMIZER? KLINGT SPANNEND...

```
1|          /*----------
2|           * Estimate number of main-table pages fetched, and compute I/O cost.
3|           *
4|           * When the index ordering is uncorrelated with the table ordering,
5|           * we use an approximation proposed by Mackert and Lohman (see
6|           * index_pages_fetched() for details) to compute the number of pages
7|           * fetched, and then charge spc_random_page_cost per page fetched.
8|           *
9|           * When the index ordering is exactly correlated with the table ordering
10|          * (just after a CLUSTER, for example), the number of pages fetched should
11|          * be exactly selectivity * table_size.
```

- Ja, das Beispiel ist einfach
- Ja, man muss trotzdem etwas zum Thema wissen
- Aber: **User** und **Support** haben eine **Chance** !

LIEBE ZUM DETAIL

UM DAS LEBEN FÜR ALLE ZU VERBESSERN

KÜRZLICH: MEHR ALS 1000 WORTE

```
test=# SELECT 'CREATE TABLE tab' || x || ' (id int) '
       FROM generate_series(1, 3) AS x;
          ?column?
----------------------------
 CREATE TABLE tab1 (id int)
 CREATE TABLE tab2 (id int)
 CREATE TABLE tab3 (id int)
(3 rows)

test=# \gexec
CREATE TABLE
CREATE TABLE
CREATE TABLE
```

Kein Söldner der Welt macht so etwas

ABSCHLIESSEND

EIN PAAR GEDANKEN

Erleuchtun

“FREEDOM HAS MANY DIFFICULTIES
& DEMOCRACY IS NOT PERFECT,
BUT WE HAVE NEVER HAD TO PUT
A WALL UP TO KEEP OUR PEOPLE IN”

WER EINE GUTE LÖSUNG
HAT,
MUSS NICHT EINSPERREN

ANY QUESTIONS?

ASK ANYTHING

HANS-JÜRGEN SCHÖNIG

CEO & FOUNDER

EMAIL

hs@cybertec-postgresql.com

PHONE

+43 2622 930 22 - 666

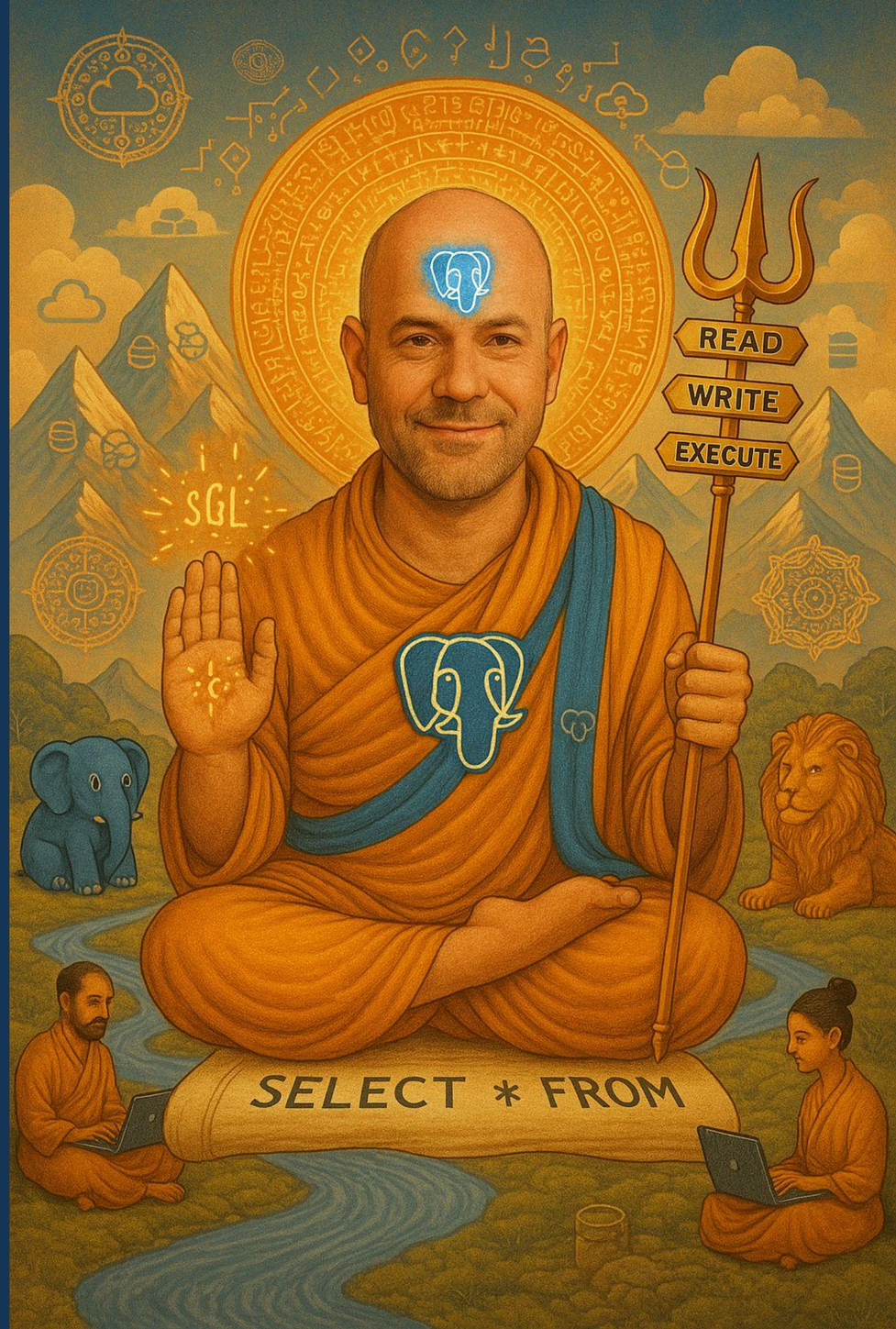
www.cybertec-postgresql.com

@cybertec-postgresql

www.youtube.com/@cybertecpostgresql

PROUD CONTRIBUTOR TO

PostgreSQL
the world's most advanced open source database



→ Tickets now live!

→ Call for papers

→ Call for sponsors

pgday.at

Open Alliance

For PostgreSQL Education

- Independent industry certification for PostgreSQL
- First launch at PGConf.DE 2025
- For more information visit booths of CYBERTEC, Data Egret and Proventa

Scan for Updates

oapg-edu.org

MEET OUR PARTNERS AT PGCONF.DE