

A look at the Elephants Trunk

PostgreSQL 16

PostgreSQL@SCaLE 2023
Pasadena, USA

Magnus Hagander
magnus@hagander.net



Magnus Hagander

- Redpill Linpro
 - Principal database consultant
- PostgreSQL
 - Core Team member
 - Committer
 - PostgreSQL Europe

PostgreSQL 16

Development schedule

- June 2022 - branch 15
- July 2022 - CF1
- September 2022 - CF2
- November 2022 - CF3
- January 2023 - CF4
- March 2023 - CF5 (ongoing!)

New features

- DBA and administration
- SQL and developer
- Backup and replication
- Performance

Breaking changes

Windows

- Minimum version now Windows 10

Removed ports

- HPUX
- Itanium

promote_trigger_file

- Removed!
- Use *pg_ctl_promote*
- Or *pg_promote()*

New features

- DBA and administration
- SQL and developer
- Backup and replication
- Performance

psql

- Use the extended query protocol

psql

- Use the extended query protocol

```
postgres=# SELECT count(*) FROM pg_class WHERE
postgres-#         relkind=$1 AND relname NOT LIKE $2
postgres-# \bind 'r' 'pg_%'
postgres-# \g
count
-----
      4
(1 row)
```

SYSTEM_USER

- What user was used to log in

SYSTEM_USER

- What user was used to log in

```
postgres=# SELECT CURRENT_USER, SYSTEM_USER;
current_user | system_user
-----+-----
mha          | peer:mha
(1 row)
```

pg_hba/pg_ident

- Match users and databases with regexp

pg_hba/pg_ident

- Match users and databases with regexp

```
host    /^dev    /^dev    10.0.0.0/24    scram-sha-256
host    sameuser /^test   10.0.0.0/24    scram-sha-256
```


pg_hba/pg_ident

- Include files
 - Not just include-lists

pg_hba/pg_ident

- Include files
 - Not just include-lists

```
include_if_exists pg_hba_override.conf
```

```
include pg_hba_test.conf
```

```
include_dir all_my_pghba
```

Maintenance permissions

- Grant maintenance tasks to non-table-owners
 - VACUUM, ANALYZE
 - CLUSTER
 - REINDEX
 - REFRESH MATERIALIZED VIEW
 - LOCK TABLE

Maintenance permissions

```
postgres=# GRANT MAINTAIN ON mytable TO testuser;  
GRANT
```

```
postgres=# GRANT pg_maintain TO testuser;  
GRANT
```

VACUUM

- Autovacuum instrumentation

```
frozen: 47 pages from table (35.34% of total) had 10000 tuples frozen
```

VACUUM

Page level freezing

- Freeze individual pages much earlier
- More efficient than row-at-a-time
- Also triggers after FPI by heap pruning

Statistics

- Track last scan of relation
- Separate sequential and indexes

Statistics

- Track last scan of relation
- Separate sequential and indexes

```
-[ RECORD 1 ]-----+-----  
reloid          | 16392  
.....  
seq_scan       | 13  
last_seq_scan  | 2023-03-05 17:06:23.993055+01  
...  
idx_scan       |  
last_idx_scan  |  
.....
```


pg_stat_io

- More detailed i/o statistics
- Enhances what was in *pg_stat_bgwriter*
- Global but by types

pg_stat_io

- More detailed i/o statistics
- Enhances what was in *pg_stat_bgwriter*
- Global but by types

```
backend_type | standalone backend
io_object    | relation
io_context   | normal
reads        | 689
writes       | 1008
extends      | 493
op_bytes     | 8192
evictions    | 0
reuses       |
fsyncs       | 0
```

ICU locales

- Now the default!

ICU locales

- Now the default!

```
...  
The database cluster will be initialized with this locale configuration:  
  provider:      icu  
  ICU locale:   sv  
  LC_COLLATE:   sv_SE.UTF8  
...
```

New features

- DBA and administration
- SQL and developer
- Backup and replication
- Performance

ANY_VALUE

- SQL standard aggregate
- "Any value in group"

ANY_VALUE

- SQL standard aggregate
- "Any value in group"

```
postgres=# SELECT relkind, count(*), ANY_VALUE(relname)
```

```
postgres-# FROM pg_class GROUP BY relkind;
```

relkind	count	any_value
r	70	mytable
v	141	pg_shadow
i	164	pg_toast_1255_index
t	40	pg_toast_1255

Set STORAGE on creation

- Previously required create + alter

Set STORAGE on creation

- Previously required create + alter

```
CREATE TABLE mytable (  
  a TEXT STORAGE EXTERNAL  
)
```

Non-decimal literals

- For integer and numeric
- Hexadecimal, octal and binary
- Also allow underscores

Non-decimal literals

- For integer and numeric
- Hexadecimal, octal and binary
- Also allow underscores

```
postgres=# SELECT 0x123, 0b110, 0o0664;  
          291 |          6 |          436
```

Non-decimal literals

- For integer and numeric
- Hexadecimal, octal and binary
- Also allow underscores

```
postgres=# SELECT 0x123, 0b110, 0o0664;  
          291 |          6 |          436
```

```
postgres=# SELECT 0x12_34, 0x1234, 0x_1234;  
          4660 |          4660 |          4660
```

New features

- DBA and administration
- SQL and developer
- Backup and replication
- Performance

pg_dump compression

- New syntax for specifying
 - *-Z gzip:9*
- LZ4 support
 - *-Z lz4:9*

Logical replication

Parallel apply

- Parallel workers on subscription side
- Use multiple processes for the same subscription
- Enabled per subscription

New features

- DBA and administration
- SQL and developer
- Backup and replication
- Performance

Many infrastructure

- No direct visibility
- Just runs faster
- E.g. faster memory management

Explicit SIMD support

- Intel SSE2
- ARM NEON

Parallel aggregates

- `string_agg()`
- `array_agg()`

DISTINCT using Limit

- When possible, use Limit for DISTINCT
- Instead of Unique

DISTINCT using Limit

- When possible, use Limit for DISTINCT
- Instead of Unique

```
postgres=# EXPLAIN SELECT DISTINCT relname FROM pg_class WHERE relname='test';
Limit (cost=0.27..4.29 rows=1 width=64)
-> Index Only Scan using pg_class_relname_nsp_index on pg_class (cost=0.27.
idth=64)
    Index Cond: (relname = 'test'::name)
```

GROUP BY/DISTINCT

- Smarter sort orders for aggregates
 - E.g. use an index
- Remove redundant grouping
 - E.g. multiple keys when known identical

Window functions

- Allowed to alter their own options
- Some functions don't care about ROWS or RANGE
- ROWS is faster than RANGE
- So allow functions to change it on the fly!
 - `row_number()`, `rank()`, `dense_rank()`
 - `percent_rank()`, `cume_dist()`, `ntile()`

Still in the queue

Management

- transaction_timeout

Statistics

- Further additions to `pg_stat_io`
 - IO timing
 - Shared buffers
- Checkpointer related stats
- `pg_stat_session`

Connections

- Use system CA pool
- Libpq reject unexpected auth requests
- Load balancing in libpq

SQL

- MERGE improvements
 - MERGE RETURNING
 - WHEN NOT MATCHED BY SOURCE
- SQL/JSON (dropped from 15)

Performance

- Parallel hash full join

There's always more

There's always more

- Lots of smaller fixes
- Performance improvements
- etc, etc
- Can't mention them all!

Please help!

- Snapshot version available
- Download and test!
 - apt packages available
 - rpm/yum packages available

Thank you!

Magnus Hagander

magnus@hagander.net

@magnushagander

<https://www.hagander.net/talks/>

This material is licensed



