

Major Features: Postgres 18

BRUCE MOMJIAN



POSTGRESQL is an open-source, full-featured relational database. This presentation gives an overview of the Postgres 18 release.

<https://momjian.us/presentations>



Creative Commons Attribution License

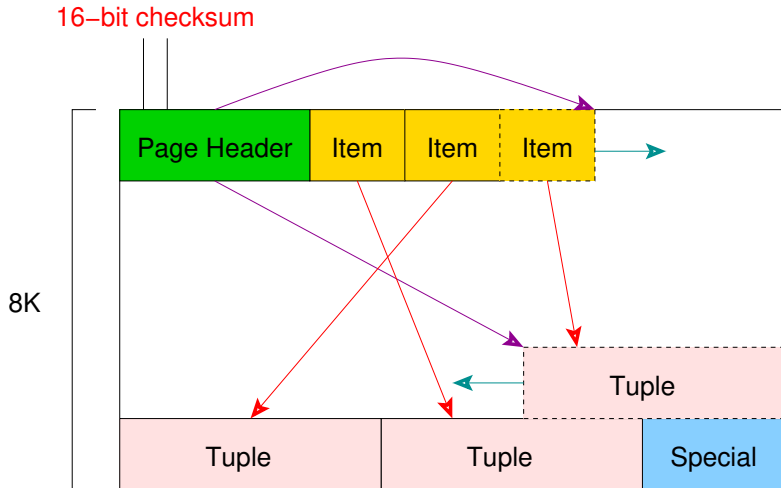
Last updated: October 2025

Postgres 18 Feature Outline

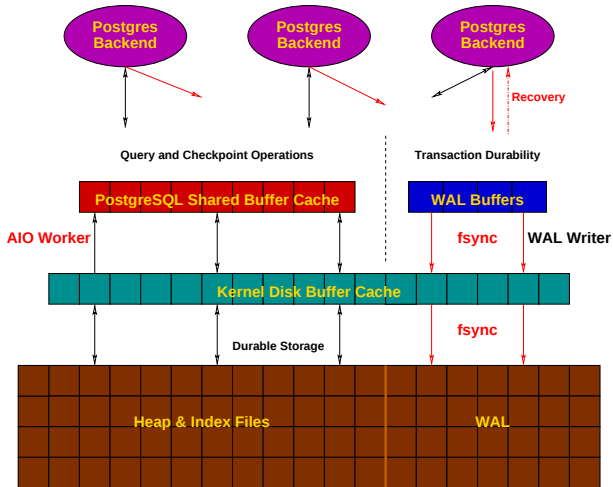
1. Checksums enabled by default
2. Asynchronous I/O workers
3. Skip scans
4. Other optimizer improvements
5. Pg_upgrade optimizer statistics preservation
6. COPY error limits
7. Foreign data wrapper SCRAM authentication

Full item list at https://momjian.us/pgsql_docs/release-18.html.

1. Checksums Enabled by Default



2. Asynchronous I/O Workers



<https://pganalyze.com/blog/postgres-18-async-io>

3. Skip Scans: Setup

```
CREATE TABLE skip_scan_test (  
    a INTEGER,  -- values 1-5  
    b INTEGER,  -- values 1-5  
    c SERIAL    -- unique  
);  
  
INSERT INTO skip_scan_test (a, b)  
    SELECT random(1, 5), random(1, 5)  
    FROM generate_series(1, 100000);  
  
CREATE INDEX i_skip_scan_test ON skip_scan_test (a, b, c);  
  
ANALYZE;  
  
\set EXPLAIN 'EXPLAIN (COSTS OFF)'
```

Unchanged

```
:EXPLAIN SELECT c  
FROM skip_scan_test  
WHERE a = 1;
```

QUERY PLAN

```
Bitmap Heap Scan on skip_scan_test  
  Recheck Cond: (a = 1)  
    -> Bitmap Index Scan on i_skip_scan_test  
        Index Cond: (a = 1)
```

```
:EXPLAIN SELECT c  
FROM skip_scan_test  
WHERE a > 5;
```

QUERY PLAN

```
Index Only Scan using i_skip_scan_test on skip_scan_test  
  Index Cond: (a > 5)
```

Unchanged

```
:EXPLAIN SELECT c
FROM skip_scan_test
WHERE a = 1 AND b = 2 AND c = 100;
          QUERY PLAN
```

```
-----
Index Only Scan using i_skip_scan_test on skip_scan_test
  Index Cond: ((a = 1) AND (b = 2) AND (c = 100))
```

-- since PG 8.1, crude skip scan

```
:EXPLAIN SELECT c
FROM skip_scan_test
WHERE a = 1 AND c = 100;
          QUERY PLAN
```

```
-----
Index Only Scan using i_skip_scan_test on skip_scan_test
  Index Cond: ((a = 1) AND (c = 100))
```

Now Index Scans, Previously Sequential Scans

```
:EXPLAIN SELECT c  
FROM skip_scan_test  
WHERE a > 1 AND c = 100;
```

QUERY PLAN

Index Only Scan using i_skip_scan_test on skip_scan_test
Index Cond: ((a > 1) AND (c = 100))

```
:EXPLAIN SELECT c  
FROM skip_scan_test  
WHERE c = 100;
```

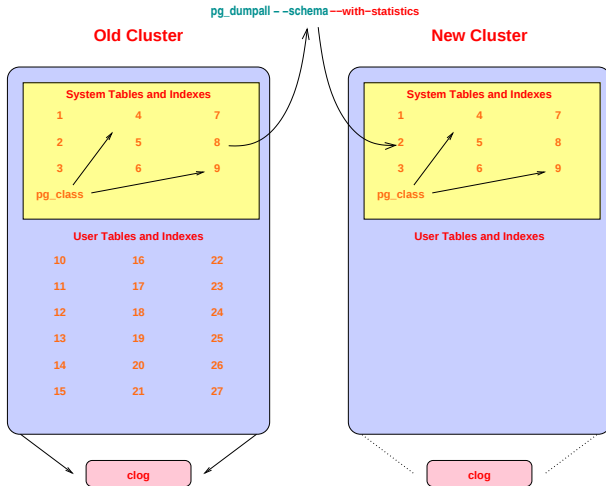
QUERY PLAN

Index Only Scan using i_skip_scan_test on skip_scan_test
Index Cond: (c = 100)

4. Other Optimizer Improvements

- Automatically remove some unnecessary table self-joins
- Convert some `IN (VALUES ...)` to `x = ANY ...` for better optimizer statistics
- Allow transforming OR-clauses to arrays for faster index processing

5. Pg_upgrade Optimizer Statistics Preservation



Also available with *pg_dump/pg_dumpall*.

6. COPY Error Limits

```
CREATE TABLE copy_test (int_col INTEGER, date_col DATE, jsonb_col JSONB);
```

```
COPY copy_test (int_col) FROM STDIN WITH (ON_ERROR ignore, REJECT_LIMIT 1);
```

```
test> 1
```

```
test> 2
```

```
test> a
```

```
test> \.
```

NOTICE: 1 row was skipped due to data type incompatibility

```
SELECT * FROM copy_test;
```

int_col	date_col	jsonb_col
1	(null)	(null)
2	(null)	(null)

COPY Error Limits

```
COPY copy_test (int_col) FROM STDIN WITH (ON_ERROR ignore, REJECT_LIMIT 1);
```

```
test> 3
```

```
test> b
```

```
test> c
```

```
test> \.
```

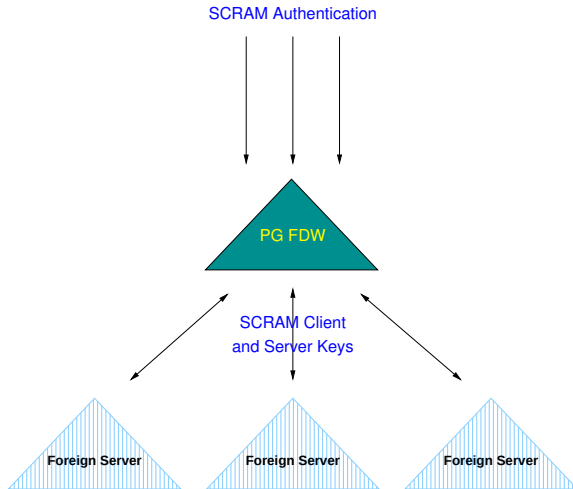
```
ERROR:  skipped more than REJECT_LIMIT (1) rows due to data type incompatibility
```

```
CONTEXT:  COPY copy_test, line 3, column int_col: "c"
```

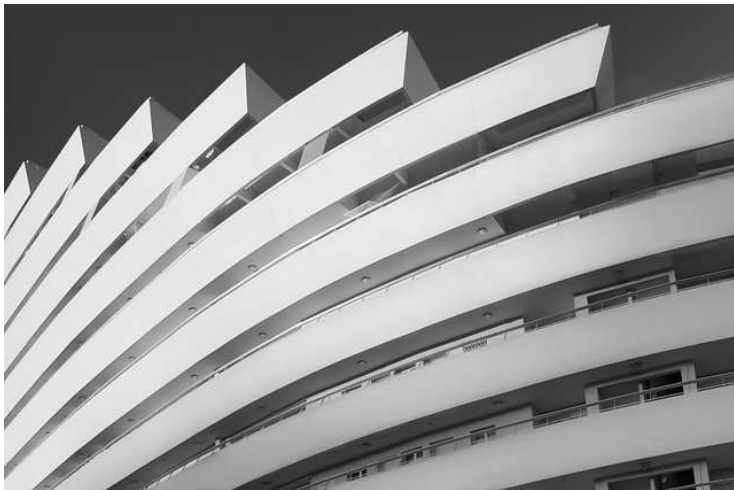
```
SELECT * FROM copy_test;
```

int_col	date_col	jsonb_col
1	(null)	(null)
2	(null)	(null)

6. Foreign Data Wrapper SCRAM Authentication



Conclusion



<https://momjian.us/presentations>

<https://www.flickr.com/photos/thomasletholsen/>