PostgreSQL 9.4

PGConf.EU 2014
Madrid, Spain

Magnus Hagander
magnus@hagander.net

PRODUCTS • CONSULTING • APPLICATION MANAGEMENT • IT OPERATIONS • SUPPORT • TRAINING

Magnus Hagander

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

Do you read...

- planet.postgresql.org

Thanks to!

- depesz
- Michael Paquier
- (others)
- And the developers of course!

Development schedule

- June 14, 2013 - branch 9.3
- June 2013 - CF1
- September 2013 - CF2
- November 2013 - CF3
- January 2014 - CF4
- July 2014 - beta2!

Current status

- In beta testing!
- Please help with testing!

Current status

- Statistics!
  - 2222 files changed
  - 131805 insertions (+)
  - 59333 deletions(-)
- Lower than 9.3
  - But..?

So what's really new

- Need some sort of categorization
- Developer and SQL features
- Infrastructure
- DBA and administration
- Replication and recovery

New features

- Developer and SQL features
- Infrastructure
- DBA and administration
- Replication and recovery

Smaller aggregate changes

- Allow variadic aggregates
  - None shipped by default
  - For user defined
- Improved EXPLAIN information

```
HashAggregate  (cost=2.94..2.98 rows=4 width=8)
  Group Key: a
  ->  Seq Scan on agg  (cost=0.00..2.11 rows=111 width=8)
```

FILTER aggregates

- Add support for FILTER clause to aggregates
- No more case-then-null!

```
SELECT a,
       count(*),
       count(*) FILTER (WHERE b > 5)
FROM agg GROUP BY a
```

Ordered-set aggregates

- New class of aggregates
  - "Offset in group"
- WITHIN GROUP
- Also Hypothetical aggregates

Ordered-set aggregates

- Most common value in group

```
SELECT a,
       mode() WITHIN GROUP (ORDER BY b)
FROM agg GROUP BY a
```

Ordered-set aggregates

- Percentiles

```
SELECT a,
       percentile_cont(0.3) WITHIN GROUP (ORDER BY b),
       percentile_disc(0.3) WITHIN GROUP (ORDER BY b)
FROM agg GROUP BY a
```

Ordered-set aggregates

- Hypothetical rows

```
SELECT a,
       rank(4) WITHIN GROUP (ORDER BY b),
       percent_rank(4) WITHIN GROUP (ORDER BY b)
FROM agg GROUP BY a
```

Improved updatable views

- Partially updatable views
- Some columns can be updated, others not
- Automatically detected

Improved updatable views

- WITH CHECK OPTION
- Only allow rows visible through view
- *LOCAL*
  - Only conditions on current view checked
- *CASCADE*
  - Recursively check on parents
  - Default once *CHECK OPTION* is specified

UNNEST

- Multi-argument UNNEST
  - Unnest multiple arrays at once

```
SELECT * FROM unnest(
    array['a', 'b', 'c', 'x', 'y'],
    array['d', 'e', 'f']
)
```

WITH ORDINALITY

- Automatic row number for unnested rows

```
SELECT * FROM unnest(
    array['a', 'b', 'c', 'x', 'y'],
    array['d', 'e', 'f']
) WITH ORDINALITY
```

pl/pgsql stacktrace

- You can now get a stacktrace!

```
CREATE OR REPLACE FUNCTION public.inner_func() RETURNS integer AS  $$
DECLARE
  stack text;
BEGIN
  GET DIAGNOSTICS stack = PG_CONTEXT;
  RAISE NOTICE E'--- Call Stack ---\n%', stack;
  RETURN 1;
END;
$$ LANGUAGE plpgsql;
```

JSONB

- "Binary json"
- Parsed JSON data
  - Current json datatype just stores text
- Basic datatyping
- Key-order not preserved

JSONB

- "hstore-style" indexes
- Nested structure support
- Containment operators

```
SELECT * FROM myjsontable
 WHERE jsondata @> '{"somekey": "somevalue",
  "otherkey": "othervalue"}'
```

JSONB

- Compact GIN indexes
- Much faster than 9.3
  - Also faster than e.g. MongoDB

New features

- Developer and SQL features
- Infrastructure
- DBA and administration
- Replication and recovery

Dynamic background workers

- 9.3 got background workers
- Only at postmaster startup
- Can now be started dynamically

Dynamic shared memory

- Shared memory can be allocated on request
- Main segment still fixed at startup
- Requested by e.g. bgworkers
- Also supports lightweight message queue

MVCC catalog access

- SnapshotNow has been removed
- All catalog access is now MVCC
- Extensions relying on it will break
  - This is intentional
- Simpler and more robust code
- Future decreased locking

Logical changeset extraction

- Fetch logical changes from WAL
- Foundation for future replication and analysis
  - E.g. the BDR project

Replication slots

- Keep track of standbys
- Automatically block WAL removal
- No more need for wal_keep_segments?

New features

- Developer and SQL features
- Infrastructure
- DBA and administration
- Replication and recovery

MATERIALIZED VIEWS

- 9.3 added materialized views
  - Limited usability due to locking
- 9.4 adds concurrent refresh

```
REFRESH MATERIALIZED VIEW CONCURRENTLY myview
```

- Requires UNIQUE index on view

Move objects in tablespaces

- Move all objects in tablespaces
  - Or all tables, all indexes, etc.

```
ALTER TABLESPACE pg_default
  MOVE INDEXES TO ssd;

ALTER TABLESPACE ssd
  MOVE ALL TO pg_default;
```

pg_prewarm

- Prewarm your cache
- Extension with pg_prewarm() function
- Prewarm OS or Postgres caches

GIN compressed posting lists

- Each entry contains "array of pointers"
  - Page number and offset
- Pointer size compressed from 90 bytes to 21
  - Store as "difference from previous"
- Up to 6x smaller indexes!

GIN Index fastscan

- "Smarter" scan order of GIN posting list
- Start with the smallest list...
  - Skip through other lists
- Big improvement for "common AND rare"

Configuration

```
ALTER SYSTEM
 SET work_mem='10MB';

SELECT pg_reload_conf();
```

ALTER SYSTEM SET

- Variables in separate config file
- Overrides what's in postgresql.conf
- Reload still required
- Contexts still applies
  - Restart can be tricky!

New configuration parameters

- autovacuum_work_mem
- Default -1 = use maintenance_work_mem
- Can now be controlled independently

New configuration parameters

- session_preload_libraries
- Loaded at session startup
- But not just from plugins directory

New configuration parameters

- wal_log_hints
- Log hintbit changes to WAL
- Required for rewind tools when not using checksums
- Hint about checksum log increments

pg_stat_statements

- pg_stat_statements now exposes query id
- Internal hash value
  - Based on parse tree
- NOT stable across versions
  - Or platforms
  - Or schema modifications (some)

New features

- Developer and SQL features
- Infrastructure
- DBA and administration
- Replication and recovery

Time delayed standbys

- Delay WAL application on slave
- Replays all normal WAL, delays at commit
- "fast recovery starting point"
- min_recovery_apply_delay=30min
  - in recovery.conf

Backup improvements

- Relocating tablespaces in pg_basebackup
- Statistics view pg_stat_archiver

There's always more

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

Tiny favorite?

- Dynamic library loading logs to DEBUG1
- Particularly useful for local_preload_libraries
- Less logspam!

Just because we're Postgres

- Date parsing now supports years >5 digits
  - ISO parsing already supported this!
  - Non-standard formats now supported as well

Thank you!

Magnus Hagander
magnus@hagander.net
@magnushagander