PostgreSQL 9.3

PGDay NYC 2013
New York City, NY

Magnus Hagander
magnus@hagander.net

Magnus Hagander

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

Do you read...

- planet.postgresql.org

Thanks to!

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

Development schedule

- June 1st, 2012 - branch 9.2
- June 2012 - CF1
- September 2012 - CF2
- November 2012 - CF3
- January 2013 - CF4

Not quite to plan

- We had some new ideas...

Not quite to plan

- We had some new ideas...
- Dec 2012 - Feature Planning Week
  - Never happened
- Feb 2013 - Final Triage
  - Never happened (or not yet)
- But things are still moving forward..

Current status

- CF4 in progress
- 57 committed patches
- 10 returned
- 33 pending
  - Clearly we need your help!

Current status

- Statistics!
  - 1889 files changed
  - 110250 insertions(+)
  - 85204 deletions(-)
- Linecount significantly lower (30%!) than 9.2
- More to come of course

So what's really new

- Big spread as usual
- DBA features
- Developer features
- Hard to categorize
- Not the same performance focus as 9.2

New features

- Developer features
- Administration features
- "Other"...

LATERAL support

- Support for LATERAL SQL queries

LATERAL support

- Support for LATERAL SQL queries
- You all know what that is, right?

LATERAL support

- Reference part of a resultset from the same level
  - Subquery or function call in FROM
- Even when dependent on other things at the same level

LATERAL support

```
SELECT c.relname, p.increment
FROM pg_class c,
     pg_sequence_parameters p(c.oid)
WHERE c.relkind='S'
```

LATERAL support

```
SELECT c.relname, p.increment
FROM pg_class c,
     LATERAL pg_sequence_parameters p(c.oid)
WHERE c.relkind='S'
```

LATERAL support

- Even easier
- LATERAL applied automatically for function calls
- But still required for subselects

Range type awesomeness

- You all use range types now, right?
- One of the best features in 9.2!

Range type awesomeness

- SP-GiST support
  - You probably want this!
- Statistics for range types

JSON enhancements

- 9.2 added support for JSON datatype
- Didn't actually do much
- Required pl/v8 to be useful
- Incremental enhancements

JSON enhancements

- json_agg() aggregate
  - Aggregate any query into a single JSON value

```
WITH q AS (SELECT whatever...)
SELECT json_agg(q) FROM q
```

JSON enhancements

- Mappings from hstore
- hstore_to_json()
- Also available as a cast
- Especially useful in non-hstore aware clients!

array enhancements

- New array_remove() and array_replace()
- Modify contents of array

```
SELECT array_remove('{a,b,c}'::text[], 'b')
```

- Still does full value update
- But simplifies queries

Lots of VIEW work

- Nothing truely revolutionary
- But lots of making life easier

Auto updateable views

- We've "always" done updateable views
- But you had to create your own RULE's.
  - Or triggers in 9.1 and up
- No need anymore for simple views
  - Those defined by SQL-92 as updateable
- You can still create your own for advanced views

Recursive views

- Mostly syntactic sugar
- CREATE RECURSIVE VIEW
- Instead of combination of CREATE VIEW and WITH RECURSIVE
- End result is the same

Materialized views

- A VIEW
- That materializes a snapshot of the data in it
- Snapshot is queried instead of main view
- (supposedly) cheaper and faster
  - Assumes the underlying VIEW is costly
  - Can have custom indexes
- We used to have to do this manually with trigggers...

Materialized views

```sql
CREATE MATERIALIZED VIEW myview
AS SELECT a,b FROM mytable WHERE x < 100
```

```sql
SELECT * FROM myview
```

Materialized views

- Manual refreshing
  - So far
  - You can use cron of course
- No incremental refresh
- Locking issues during refresh

New features

- Developer features
- Administration features
- "Other"...

COPY FREEZE

- Ability to insert pre-frozen data
  - No need to revisit tuples later (e.g. VACUUM)
  - Unless modified
- Only into newly created tables
  - Or truncated
  - Same transaction
  - E.g. a partition...

COPY PIPE

- Send and receive COPY data from program instead of file
- E.g. for gzipped files

```
postgres=# COPY t FROM PROGRAM 'zcat /tmp/t.csv.gz'
```

Foreign Data Wrappers

- Access data in remote databases
- Plenty available in 9.2:
  - csv files
  - Oracle
  - MySQL
  - Redis
  - MongoDB
  - ODBC
- Quite embarrassing, really...

postgres_fdw

- Finally!
- No more dblink required (still there!)
- Access remote PostgreSQL servers "properly"
- Supports remote cost estimates
- Pushes down quals (when possible)
- Foundation for even more optimizations

Writeable FDWs

- Ability to update foreign tables
- INSERT/UPDATE/DELETE
- Transaction aware (of course)
- Can be slow for complicated updates/deletes

Replication

- First release in a long time without major replication features
- But there has to be something?

Follow timeline changes

- 9.2 and earlier only followed timeline changes using log archive
- 9.3 will work on streaming only replication
- Required to "repoint slave"
- But surely you already had a log archive?

Background worker processes

- Infrastructure support for background workers
- Nothing really built on it yet
- Write your own plugins!

No more SysV shared memory

- Sick and tired of sysctl?
- Probably don't need tihs anymore!

pg_ident.conf parsed on reload

- We did this for everything else
- Last one to fall
- Safer config-edits!

Page level checksums

- Each datapage now has a 16-bit checksum
- Early detection of data corruption
- Enabled at initdb time
- Comes at a cost...

Logging enhancements

- Object names are now logged in separate fields
- Easier to extract in client drivers
- No need to parse text messages
  - Which might be different in different locales

Logging enhancements

- Log line number and contents of HBA row for failed logins
- Easier to track down incorrect HBA files
- Because we all made that mistake...

New foreign key locking

- New lock levels
  - FOR KEY SHARE
  - FOR NO KEY UPDATE
- Much better concurrency for FOREIGN KEYs

Event triggers

- Sometimes referred to as DDL triggers
- Potentially support for more later
- Global per database
  - Table name (or other) as argument

There's always more

- Lots of smaller fixes
- For example, many around performance
- Can't mention them all!

Things still in the queue

- We're still in CF4...

Things still in the queue

- Extension templates
- Parallel pg_dump
- Index based regexp search (pg_trgm)
- SET PERSISTENT
- Logical changeset generation

Thank you!

Magnus Hagander
magnus@hagander.net
@magnushagander